OBJECTIVE MEASUREMENT OF FABRIC SOFTNESS AND PILLING USING HAND CRAFTED FEATURES AND DEEP LEARNING

A THESIS SUBMITTED TO
THE GRADUATE SCHOOL OF INFORMATICS INSTITUTE
OF
MIDDLE EAST TECHNICAL UNIVERSITY

BY

SEYMUR MAMMADLI

IN PARTIAL FULFILLMENT OF THE REQUIREMENTS
FOR
THE DEGREE OF MASTER OF SCIENCE
IN
MODELLING AND SIMULATION

DECEMBER 2021

Approval of the thesis:

OBJECTIVE MEASUREMENT OF FABRIC SOFTNESS AND PILLING USING HAND CRAFTED FEATURES AND DEEP LEARNING

submitted by **SEYMUR MAMMADLI** in partial fulfillment of the requirements for the degree of **Master of Science in Modelling and Simulation Department, Middle East Technical University** by,

Prof. Dr. Deniz ZEYREK BOZŞAHİN
Dean, Graduate School of **Informatics Institute** _____

Assoc. Prof. Dr. Elif Sürer
Head of Department, **Modelling and Simulation** _____

Prof. Dr. Ahmet Oğuz Akyüz
Supervisor, **Computer Engineering, METU** _____

Examining Committee Members:

Prof. Dr. Alptekin Temizel
Modelling and Simulation, METU _____

Prof. Dr. Ahmet Oğuz Akyüz
Computer Engineering, METU _____

Assoc. Prof. Dr. Elif Sürer
Modelling and Simulation, METU _____

Assoc. Prof. Dr. Erdem Akagündüz
Modelling and Simulation, METU _____

Assist. Prof. Dr. Aslı Gençtav
Computer Engineering, TED _____

Date: 09.12.2021

I hereby declare that all information in this document has been obtained and presented in accordance with academic rules and ethical conduct. I also declare that, as required by these rules and conduct, I have fully cited and referenced all material and results that are not original to this work.

Name, Surname: Seymur Mammadli

Signature :

ABSTRACT

OBJECTIVE MEASUREMENT OF FABRIC SOFTNESS AND PILLING USING HAND CRAFTED FEATURES AND DEEP LEARNING

Mammadli, Seymur
M.S., Department of Modelling and Simulation
Supervisor: Prof. Dr. Ahmet Oğuz Akyüz

December 2021, 77 pages

Fabric softness is a complex tactile sensation perceived by the user even before the fabrics are worn. Softness is usually the property of surface perceived by touching or pressing a finger on the fabric surface. Fabric friction properties significantly affect the tactile sensation of the garments. The yarn used, the finishing works, and the fabric structure (weaving, knitting, etc.) affect the softness. In addition, the hardness of the water used during washing, washing movements, the amount and content of the detergent and softener used also have permanent effects on the fabric softness. Softness can be evaluated by the jury members with proven effectiveness according to the predetermined scale. Our achievement within the scope of the thesis is to eliminate the differences that may occur as a result of the subjective evaluation, which may arise from qualitative observations by basing the degree of softness evaluated qualitatively on numerical data and to obtain clearer and more precise results by adding quantitative features to the evaluation process. The methodology developed for softness assessment is also applied for another textile deterioration parameter, namely pilling, and its results are also reported.

Keywords: Deep learning, machine learning, hand crafted features, textile features

ÖZ

MANÜEL ÖZNİTELİK ÇIKARIMI VE DERİN ÖĞRENME KULLANILARAK KUMAŞ YUMUŞAKLIĞI VE BONCUKLANMA DEĞERLERİNİN OBJEKTİF BİR ŞEKİLDE ÖLÇÜLMESİ VE SINIFLANDIRILMASI

Mammadli, Seymur
Yüksek Lisans, Modelleme ve Simulasyon Bölümü
Tez Yöneticisi: Prof. Dr. Ahmet Oğuz Akyüz

Aralık 2021 , 77 sayfa

Kumaş yumuşaklığı kumaşların giyilmesinden bile önce kullanıcı tarafından algılanan karmaşık bir dokunma hissidir. Yumuşaklık genellikle kumaşın parmaklarla sıkılması veya preslenmesi ile algılanan yüzey özelliğidir. Kumaş sürtünme özellikleri, giysilerin dokunma duyumlarını büyük ölçüde etkiler. Kullanılan iplik, bitim işleri ve kumaş yapısı (dokuma, örme vb.) yumuşaklığı etkilemektedir. Bunun yanında yıkama sırasında işlem gördüğü su sertliği, yıkama hareketleri, kullanılan deterjan ve yumuşatıcının miktarı ve içeriğinden de etkilenmektedir. Görsel olarak test edilen bir diğer tekstil özelliklerinden olan yumuşaklık, etkinliği kanıtlanmış jüri üyeleri tarafından aşağıdaki skalaya göre değerlendirilebilmektedir. Tez kapsamındaki kazanımımız nitel olarak değerlendirilen yumuşaklık derecesinin, sayısal verilere dayandırılarak, nitel gözlemlerden doğabilecek görsel değerlendirme sonucu oluşacak farklılıkların giderilmesi ve değerlendirme prosesine nicel özellik kazandırarak daha net ve kesin sonuçların elde edilmesidir. Yumuşaklık için geliştirilen metodoloji değerlendirme aynı zamanda başka bir tekstil bozulma parametresi, boncuklanma için de uygulanmış ve

sonuçları raporlanmıştır.

Anahtar Kelimeler: Derin öğrenme, Makine öğrenmesi, manuel öznitelik çıkarımı, tekstil özellikleri

To my family

ACKNOWLEDGMENTS

I want to express my sincere gratitude to my advisor Prof. Dr. Ahmet Oguz Akyuz, for his kind patience, guidance, encouragement, support, and suggestions through the development of this thesis work. I also thank him for his belief in me even when I lost my ambition for writing this thesis. Throughout this study, the relationship between him and me forces me to work harder and harder every day to develop quality work for his standards.

I would also express my gratitude to each of the jury members Elif Sürer, Tolga Kurtuluş Çapın, Alptekin Temizel, Ahmet Oğuz Akyüz, Zaur Taghiyev, for reading the thesis and feedback their valuable suggestions and comments.

I also thank the Arçelik team, especially to Zehra Ülger, Nur Dilara Öztürk, Engin Bayatlı, for their participation and strong support for this research. Moreover, I thank Arçelik team members Duygu and Mustafa, who repeatedly spent preparing the new datasets.

Furthermore, my deepest thanks belong to my family, who encourage me during the research. I would especially like to express my sincere gratitude to my father, who provided continuous morale and encouragement during my research. Because of his academic background, he always pushed me forward and showed me a bright future.

I am sure it will not be enough to say that I am thankful to everyone who helped me develop this thesis throughout all these years.

The numerical calculations reported in this thesı were partially performed at TUBITAK ULAKBIM, High Performance and Grid Computing Center (TRUBA resources).

TABLE OF CONTENTS

LIST OF TABLES

TABLES

LIST OF FIGURES

FIGURES

LIST OF ABBREVIATIONS

CV	Computer vision
DL	Deep learning
GLCM	Grey-Level Co-Occurrence Matrix
ML	Machine learning
HVS	Human visual system
HCF	Hand crafted features
SOTA	State of the art
TSA	Textile softness analyzer
GUI	Graphical user interface

CHAPTER 1

INTRODUCTION

1.1 Motivation and Problem Definition

Fabric softness is usually the surface property perceived by finger pressing or touching the surface of the fabric. It is a complex tactile sensation perceived by the human perception system even before the materials are contacted. Fabric friction properties significantly affect the tactile feel of the garments. The quality of the textile used, the finishing works, and the fabric structure (weaving, knitting, etc.) affect the softness characteristic of the fabric. In addition, the hardness of the water used during washing, the washing movements, the amount and content of the detergent and softener used also have permanent effects on the fabric softness. It is not easy to quantitatively measure the value of the softness of the fabric. It is one of the textile properties that could be tested visually. The jury members can evaluate the value of the softness with proven effectiveness according to the predetermined scale. This is one of the mainly used traditional methods in textile properties analysis. The problem with this method is that the vision ability of the jury members could be different from each other. Furthermore, the jury members have not been able to perfectly measure the value of softness of the fabric by only relying on the human visual system (HVS).

It is not easy to precisely define the physical difference between the fabrics displayed in the Figure 1.1, without physically interacting with the samples. It is harder for the value of the softness of the fabric. Close inspection of textile samples indicates there could be computable variables that may predict a sample's softness.

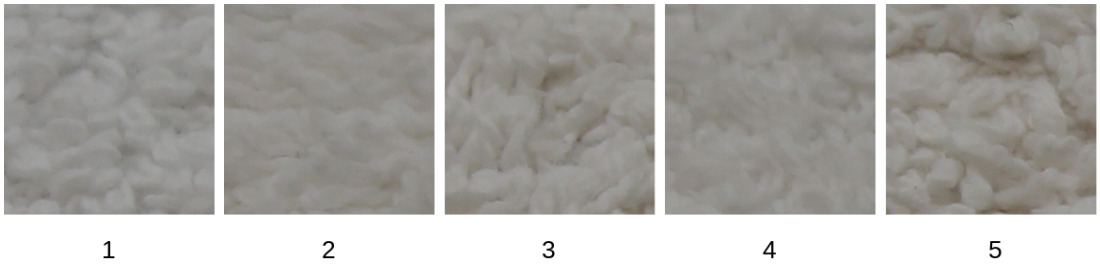


Figure 1.1: Average expert scores according to softness values display on the bottom of each image. Score 1 means the softest fabric in the value. As the value of the score increases, the value of the softness decreases.

It is not very hard for the pilling samples to differentiate two different samples through the pilling value. As shown in Figure 1.2, it is easy to determine low pilled and severe pilled pieces directly through HVS. It is easy because we have an exact definition of the pilling in our minds. Furthermore, the pilling effect on the fabric's surface is more noticeable than the softness effect on the fabric elements because of the final looking of the surface fractions. The pilling on the surface leads to surface changes even in a low amount of pilling.

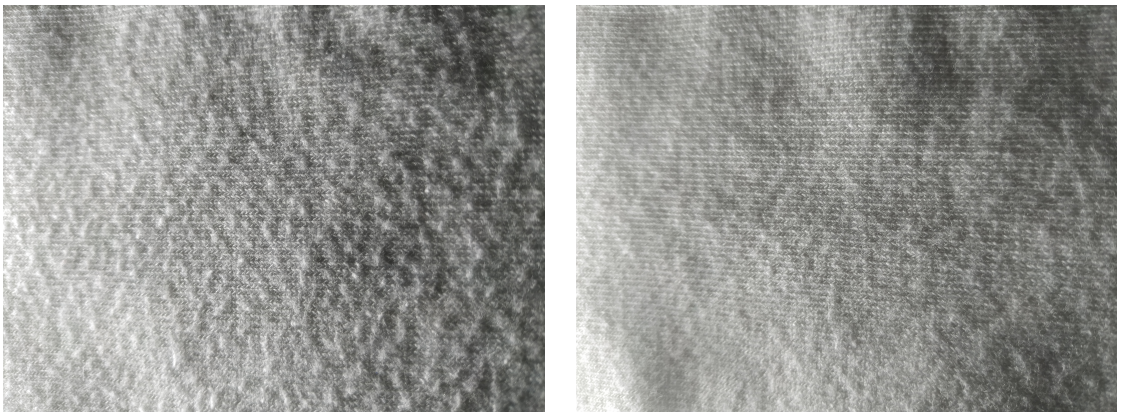


Figure 1.2: Very severe pilled and low pilled fabric samples.

In Figure 1.3, it is noticeable that even the color of the sample may change when there is a substantial amount of pilling on the surface.

Figure 1.3: Sock example with different amounts of the piling on the surface. We received this example from the Arcelik team as a reference at the beginning of the study.

Our aim within the scope of the study is to eliminate the differences that may occur as a result of visual evaluation. Furthermore, the measurement process of the softness and pilling has been automated by reducing the amount of work and resources needed for the procedure. The research aims to get more precise results by adding quantitative features to the evaluation process. As the output of the thesis, we aimed to create an image processing and machine learning (ML) methodology that provides measurement on a scale of 1-5 for both automated softness and pilling assessment.

Our motivation in this work is to understand how effectively these textile parameters

can be found by quantitative evaluation alone. This may lead to future applications such as automatically controlling the detergent and softener use within a washing machine by automatically analyzing the fabric features before starting the washing procedure.

1.2 Proposed Methods and Models

In this thesis work, two mainstream methods have been proposed to define softness and pilling of the textile by examining simple images. The main intention of the research is to implement both strategies and analyze the results.

The first method is evaluating the efficacy of the fabric using hand-crafted features (HCF), which is traditionally used in the field. In this part, four different feature sets have been used as the basis to analyze the features of the textile. The first feature set is the set of features that have been proposed by Tamura et al. [1]. The second set of features are the GLCM [4] features. Accordingly, the fourth and final feature sets are basic features and wavelet features (energy).

Furthermore, some standard and well-known computer vision (CV) based image analysis methods and mathematical methods have been tested to get a consistent measure method for the softness and pilling value of the fabric. The correlation analysis has been used to analyze the correlation between HCF analysis output and the jury evaluation output. Linear regression, Logistic regression, and Support Vector machines have been utilized to predict the class labels using a combination of the most promising features.

The second method is to use the deep learning (DL) methods to evaluate the textiles' softness and pilling. The main aim of using this method is to develop DL based classification model to classify the fabric input images based on the value of the softness and pilling. The input of the models is the input images that have been provided by the Arcelik team, which were scored by subjective scoring of jury members.

1.3 Contributions

The contribution of this thesis work is to automate the evaluation procedure of measuring the value of fabric softness and pilling which requires manual labor while using traditional methods. Through this work, the evaluation process can be done without any human interaction. Our contributions involve testing both HCF and DL features for solving this problem, comparing their effectiveness, and advancing the state-of-the-art for automatic assessment of textile parameters using only photographic images.

1.4 Novelties

The main novelty of this study is that it is one of the first work in the field that quantitatively and qualitatively evaluates fabric softness and pilling degree by applying traditional computer vision methods and DL. Some other works measure other features such as the pilling amount, wrinkle degree of the fabric. Still, none of the previous works quantitatively measures the softness value of the materials as a parametric value. Another novelty is the amount of data that we have used to fine-tune pre-trained DL models. We have used a minimum amount of data to generate our datasets and train deep-learning models. Furthermore, quantitatively measuring the fabric softness value in this research has been implemented without using any unique setup or environment. We have only used input images that have been taken by the standard camera or camera of a mobile device.

1.5 The Outline of the Thesis

Chapter 1 includes motivation and problem definition, which is followed by proposed methods and models in this research. At the end of the chapter, the contributions derived in this research and novelties of the study are presented.

Detailed literature surveys of related works are presented and explained in Chapter 2. Similar academic studies on textile feature analysis, such as research on pilling

detection, wrinkle assessment, and defect analysis using CV and ML methods, are reviewed. In Chapter 3, the theory of the methodology that we adopted and the proposed solution to the problem are explained. The details of the algorithms are also presented in this chapter.

In Chapter 4, the details of the implementation procedure and the technical details of the algorithms are explained. The details of the data collection and processing are also presented in Chapter 4. In Chapter 5, the results derived from this research are presented, and the details are explained.

Chapter 6 concludes the thesis by presenting a general overview of the research. Limitations and future works are also identified in Chapter 6, followed by several use cases.

CHAPTER 2

LITERATURE REVIEW

Measuring the fabric features' value and analyzing a characteristic of the textiles is not a new concept. The early methods are mostly based on subjective evaluation, which has drawbacks that we have stated in different paragraphs. The algorithm-based fabric analysis has been interested in many studies by the agency of improving the CV and ML field. There is a large body of research, especially on fabric defect detection, quality analysis, pilling grade analysis and pilling assessment, wrinkle analysis, and much more. Most of the study aims to improve fabric quality and better analyze fabric before production. Before CV and automated computer-based methods, all these procedures and analyses were held by experts working in the field by relying on the human perception system. Nevertheless, the human perception system is very different on different subjects. The GUI is given to the Arcelik team to be used in their facility.

2.1 Computer Vision and Machine Learning Based Fabric Feature Analysis

With the improvements of the computers' computing power and the development of new algorithms and methods in computer vision and ML, the fabric features' analysis process is shifting from traditional physical methods to digital computer-based methods. Many academic works focused on fabric feature analyses on different aspects by applying computer-based methods.

Stojanovic *et. al* [11] presents an automated vision-based system to check the quality of fabric textile. The system consists of special hardware and software that checks fabric quality by applying image processing techniques on the live data.

The authors of the [9] have developed a method to evaluate the yarn's quality in fabric using Image Processing Techniques. They proposed derivation of the parameters that define the yarn's quality along with the length and the diameter of the yarn. Furthermore, they suggest the mathematical model defines the fabric's quality based on the quality determined for the yarn and quantitative evaluation index for the yarn quality. The method applies traditional CV algorithms such as threshold, segmentation, and feature extraction on the yarn image to determine the yarn's quality.

2.1.1 Wrinkle Assessment

A wrinkle is the shape deformation on the fabric's surface after washing or during frequent usage. It directly depends on the characteristics of the fabric that the garment is made out of. Like other fabric-based phenomenons, there is much scientific research on wrinkle analysis too. Most of these studies focused on the fabric's quality—wrinkle assessment in the traditional methods made using the human perception system. Experts decide on the wrinkle score of the fabric by using subjective opinion by relying on these judges' vision abilities. The final result of this experiment depends on the personal vision ability [49, 50] of the subjects with different perception qualities. As of today, some CV and ML-based methods have focused on an objective scoring of fabric's wrinkle-like-hood [51, 52, 53]. The techniques that these methods use are very different from each other. The assessment environment is also different from simple image analysis to the special environment and special tools. Some methods use light sources from a specific angle to reflect the light on the fabric's surface, making it easy to analyze the wrinkle lines or shapes. However, this method is not suitable for textured fabrics with very sharp color changes. In their study [51] Mirjalili and Ekthiyari have suggested a special environment based on image processing to derive wrinkle assessment using known image processing algorithms objectively. The method also uses a light source to easily get the fabric's surface shape from the other side, making it possible to analyze the wrinkle easily. The fabric moves under the light source to analyze the whole patch. After that, several CV algorithms such as filtering and quantization were done to get the textile's wrinkle degree.

2.1.2 Defect Analysis

Defect detection of the fabric is the main step of the quality check of the fabric. The defect on the fabric in the industrial process leads to losing an important business profit and time. 85% of the defects that occur in the production process are related to fabric defects [40]. Defect detection is one of the main focuses of fabric inspection during manufacturing. It needed many resources and hard work to inspect fabric defects using traditional methods such as using an HVS to detect defects during the manufacturing process. As stated in the above section, the subjective inspection's success rate is based on the subjects' visual ability quality and is not the same for all subjects. Thus it is important to evaluate defect detection by using objective analysis such as computer-based automated methods [43, 44, 45, 46, 47, 48]. These methods also reduce an important amount of time and resources to inspect fabric defects in the production process. Like other fabric-based phenomena, there are many types of research on automated fabric defect detection using different computer-based methods. These methods and studies are mainly based on CV and ML-based algorithms. There is no doubt that this is because of the growth of these methods in the field.

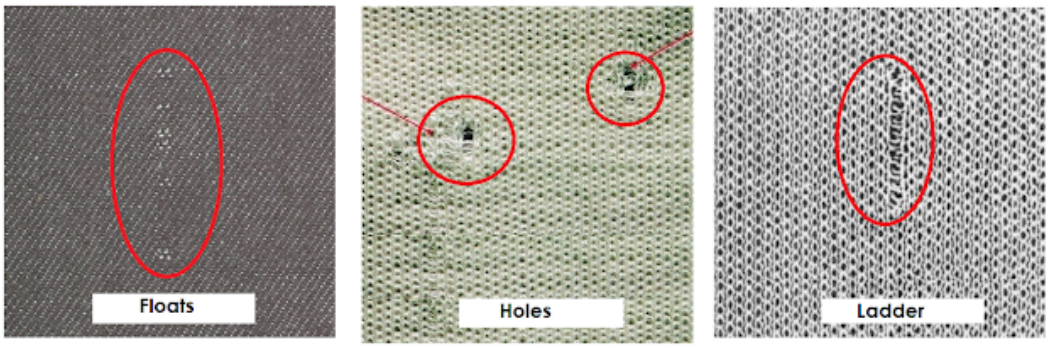


Figure 2.1: There are many types of fabric defects. As it is a research topic of many studies, it has advanced terminology and special methodology. At 'Standard Fabric Defect Glossary ' [42] there is a brief classification of many types of fabric defects and terminology used for these defects. Figure taken from [41].

2.1.3 Pilling Analysis

The pilling is textile deformation or textile defect that occurs on the fabric's surface even before the delivery of the textile to the consumer. Pilling occurs as spherical yarn pellets of different sizes and without any pattern. The main reason for the pilling occurrence is the abrasion that the fabric's surface is exposed to. It causes an unpleasant appearance on some parts or the whole surface of the garment. Besides the unpleasant looking of it, it also weakens the yarn quality in the fabric. It is also not possible to completely solve the problem after it occurs. An example of typical pilling has been displayed in Figure 2.2.

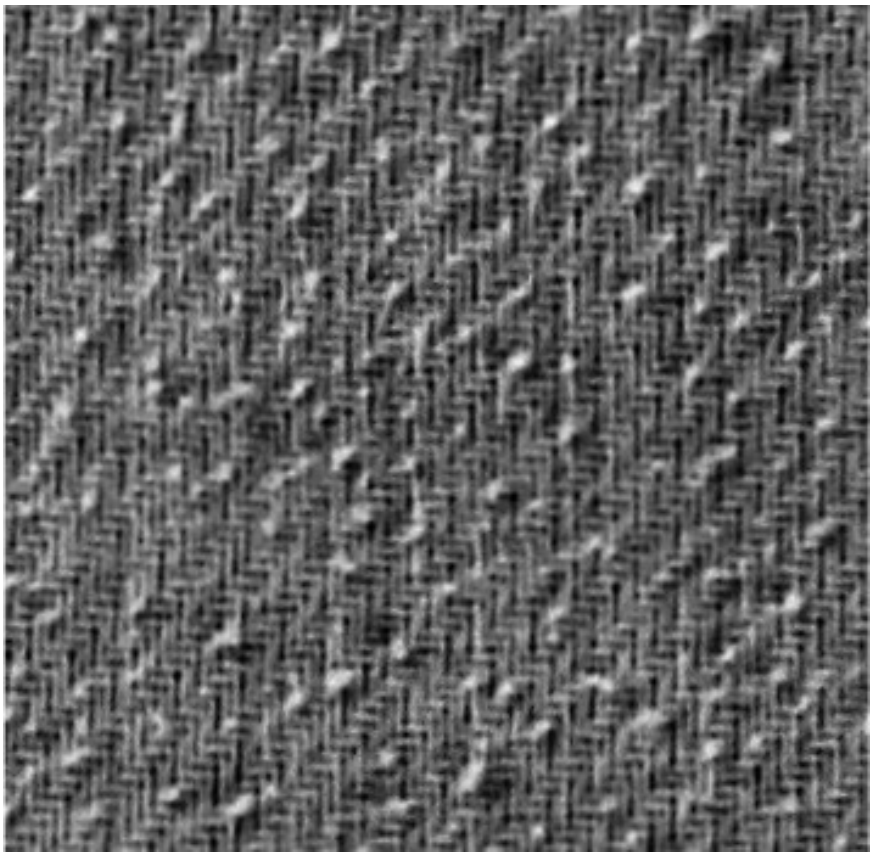


Figure 2.2: The typical pilling on the surface of the fabric. Figure taken from [19].

There are many scientific and academic studies on pilling to understand its cause and nature to increase fabric quality. It was demanding to test fabric pilling responses for the textile industry. Some unique methods physically measure the fabrics' pilling grades by doing specialized tests in a laboratory environment [20, 21].

2.1.3.1 The Martindale Abrassion and Pilling Tester Machine

It is essential to get better quality on fabrics by analyzing them before fabric production. The are many industrial methods to test fabric quality and characterizations. One of the most known traditional fabric pilling feature analysis methods is the analysis held utilizing the device named Martindale [36, 37]. Today, the Martindale equipment is also widely used in textile production companies to do experiments on textiles. The device's main purpose is to measure the durability of the fabric. It is used primarily to test textile abrasion and pilling state under special conditions. This test, also known as a rub test, aims to test the fabric's suitability for different use.

The fabric stretched on the circular lover plates in the bottom part of the device. There are small movable discs with special material such as worsted wool or some hard material covered on them. These discs continuously moved on the big plates on which the fabric stretched. The test ends when there is a noticeable change on the surface of the fabrics on the plates—the result of the experiment scoring of the multiples of 1000. The higher the value is, the fabric is more suitable to the harsh environment or harsh usage. The test conditions and parametric values make it easy to categorize the fabric tested in the experiment: The typical Martindale abrasion and pilling tester machine have been displayed at Figure 2.3.

Figure 2.3: The Martindale Abrassion and Pilling tester machine. The machine has movable small discs on different sizes which are also optimizable in size and characteristics. Figure taken from [38].

There are many CV and ML methods implemented to evaluate pilling characteristics

of the fabrics [22, 23, 24, 25, 26].

2.1.4 Softness Definition and Evaluation

Softness is one of the textile's main features, defined as the key to a comfortable experience. The standard use of the garment experience effect of this feature continuously while wearing it. Sensing and judging the softness value of fabric is a subjective process that results from textile and skin interactions. Since subjective evaluation has its problems, the focus has changed towards an objective evaluation of the fabric softness as like other characteristics. Many fabric evaluation systems and special devices have been developed to study and evaluate the softness and other characteristics of the fabric [16, 17, 18] .

Kawabata Evaluation System is a series of instruments used to measure those textile material properties that enable predictions of the aesthetic qualities perceived by human touch [39].

Another device that is one of the most used to evaluate fabric softness is the Textile Softness Analyzer (TSA) machine displayed in Figure 2.4.

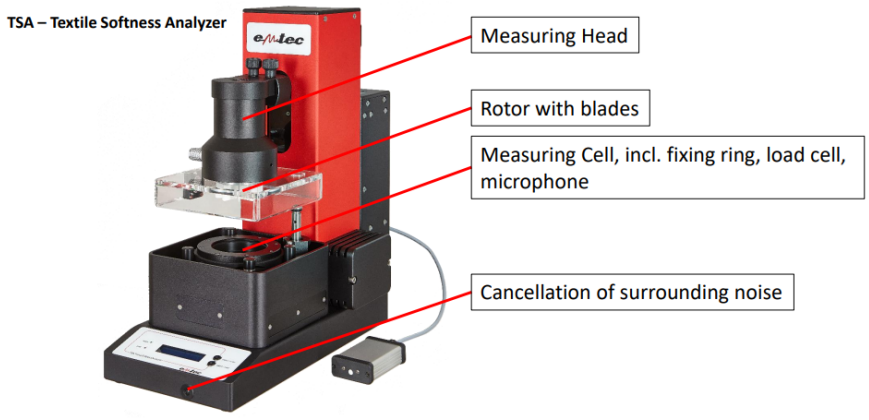


Figure 2.4: The textile softness analyzer machine. Figure taken from [12].

The TSA machine calculates three basic parameters: the softness, roughness, and stiffness of textiles and nonwovens [13]. These parameters together determine the human feeling on the fabric. It is possible to measure hand feel value by using these

parameters. Furthermore, it is also possible to measure viscoelastic, elastic, and plastic properties.

There are also other early method [14], which uses a sound sensor and detector that detects and indicates sound level frequencies directly related to the fabric's softness.

However, most of the systems have some drawbacks of its application universally either from the intricacy and complexity point of view or cost parameters of the instrument or some preferences to its application.

2.1.5 Fabric Texture and Pattern Analysis

Involvement of the computer vision in textile feature analysis is also highly applied to fabric texture analysis. There are many studies on automated fabric texture analysis based on both CV and ML. Some use the texture features to measure the quality parameters and detect the quality issues on production. In [32] the author uses a simple image segmentation method to analyze the textures of the fabric to detect fault and defects.

In [33] the authors propose a new method to measure the texture characteristics of the woven fabrics. The technique is a computer vision-based inexpensive implementation that measures weave repeat and yarn counts and the surface roughness characteristics of the texture. The authors purpose the indicator for the surface roughness, which is calculated from 2D fast Fourier transform of a 3D surface scan. The method proposed to validate by using real woven examples and synthetically simulated by computer. The authors claim an accurate result of the technique and high-speed and reliable roughness calculation from the 3D surface data.

The authors of the [34] use a white-black co-occurrence matrix to extract the texture features of the fabric. Extracted features using on BP neural network to complete the learning process.

Khan *et. al* [35] proposed a deep learning model based on data augmentation and transfer learning to automate the classification and recognition of woven fabrics. The authors use the ResNet [72] network to extract the fabric features automatically.

2.1.6 Softeners and Anti-pilling Treatment Result Analysis

Besides fabric feature analyses, there are many studies on the effects of softeners and detergents on the fabric through every washing procedure. Some of the results from some of these studies prove that it is possible to control the fabric's softness value by using the proper amount of softeners depending on the structure and current characteristics of the fabric. In [28] the authors study the effects of chemical finishing on the pilling characteristic of the cotton knitted fabrics. Three different pretreated 100% cotton fabric tested, and it found out the tendency of the material decreases as it is oppositely correlated to the amount of the chemical finishing treatment.

CHAPTER 3

PROPOSED METHOD

As stated in previous sections, our goal is to classify five different softness and pilling values through the class values, numeric values ranging from 1 to 5. The class names and descriptions for the softness and pilling degrees are listed in Table 3.1. As for the softness, the first class is the softest, and the fifth class is the firmest or hardest considering the fabric softness. An opposite coding scheme was used for pilling where the class with label 1 means the fabric has a severe amount of pilling on the surface. The fabric with label 5 means there is no noticeable pilling on the surface.

3.1 Fabric Samples

Some examples from all classes of the softness dataset are presented in Figure 3.1. All examples are from the same dataset. As seen from the samples, even for the HVS, it is not easy to differentiate classes through softness. It is almost impossible to compare examples that are next to each other on the list, but some observations can

Table 3.1: The definition of softness and pilling values.

Class	Softness Definition	Pilling Definition
1	Very soft	Very severe pilling
2	Soft	Severe pilling
3	Medium	Moderate pilling
4	Hard	Low pilling
5	Very hard	No pilling

(a) Score 1 - Very Soft

(b) Score 2 - Soft

(c) Score 3 - Medium

(d) Score 4 - Hard

(e) Score 5 - Very Hard

Figure 3.1: Example input images from every class of the softness dataset. Score 1 means the softest fabric in the value. As the value of the score increases, the value of the softness decreases.

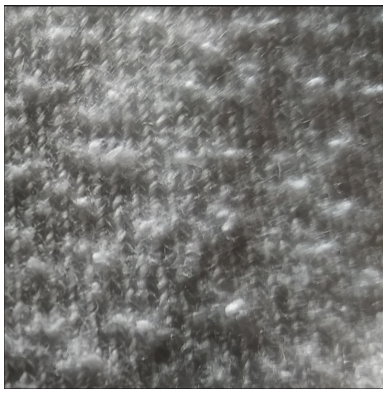
be made.

It is possible to notice from the above examples that the textile elements on the softest fabric seem smaller, and the gray values appeared because these elements' shadows are smaller in size compared to the ones which have a lower value of softness. In some examples, this is not even noticeable through the HVS. In Figure 3.1, the fourth example does not seem distinctive from the second example.

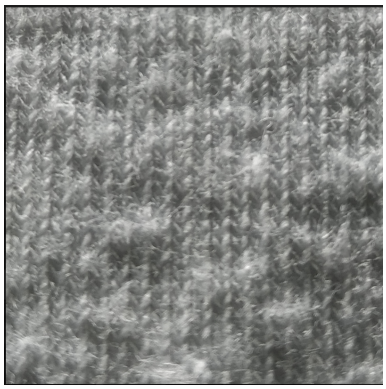
For the pilling examples (Figure 3.2), it is somewhat easier to differentiate the samples, especially for those that are at different extremes of the value range (e.g. 1 vs. 5). However, it is still not easy to differentiate the neighboring samples in the value range (e.g. 2 vs. 3). It is possible to see that on the severely-pilled sample, the pilling elements ruin the visible pattern on fabric weaving. The particles of the pilling appear to spread on the surface of the fabric. On the other hand, the weaving marks are visible for the samples with less pilling, and the yarn patterns seem tidier.

3.2 Handcrafted Features

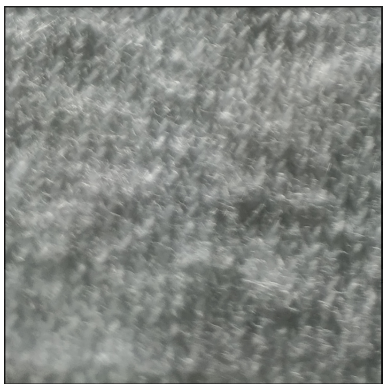
The first idea that comes to mind in image analysis is using traditional methods to analyze images' features. At the first step, we analyzed our inputs through handcrafted features (HCF) to get a robust classification algorithm. Our research made us use three different HCF-based methods to analyze the characteristics of input images. The first class of features is basic features, which are widely used in the field. Standard deviation and the mean of the images were chosen from these sets to check any correlation between these features and softness values in our patches. The second feature set is from the Tamura feature set [1] as we have mentioned above. The third feature set is GLCM [3] feature set. The details of each feature set have been explained in the below paragraphs. Furthermore, the minimum, maximum, median, and mean values of all features have been calculated and analyzed for each feature set.



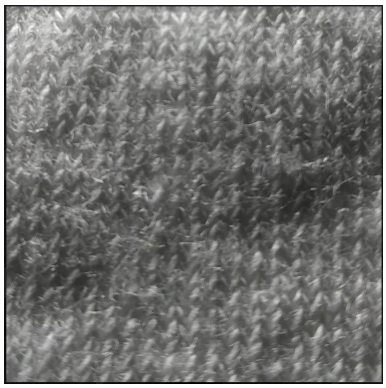
(a) Score 1 - Very severe pilling



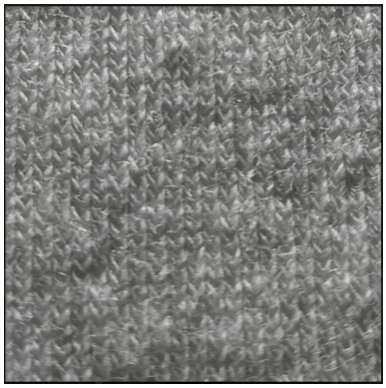
(b) Score 2 - Severe pilling



(c) Score 3 - Moderate pilling



(d) Score 4 - Low pilling



(e) Score 5 - No pilling

Figure 3.2: Example input images from every class of the pilling dataset. Score 1 means the fabric has severe pilling. As the value of the score increases, the amount of the pilling decreases.

3.2.1 Basic Features

Some simple mathematical methods calculate some basic features that sometimes tell more about image features using parametric values. Standard deviation and Mean are two basic methods that should be given helpful information about image features on image analysis. Our feature set has been included to test if there is any correlation between these features and softness values.

3.2.1.1 Standard Deviation

Standard deviation is the statistic value that measures the amount of spread on data relative to its Mean. The standard deviation σ of an image is defined as

$$\sigma = \sqrt{\frac{\sum_{i=1}^{N}(x_i - \mu)^2}{N}} \tag{3.2.1}$$

where N denotes the size of the data, x_i denotes the value of each data, and the μ is the mean value of the data.

3.2.1.2 Mean

The arithmetic mean is the sum of the elements along the axis, divided by the number of elements. The procedure can be formulated as

$$\mu = \frac{\sum_{i=1}^{N}(x_i)}{N} \tag{3.2.2}$$

where N denotes the size of the data, and x_i denotes the value of each data.

3.2.2 Tamura Features

The features categorized based on psychological studies on human perception suggested by Tamura et al. [2] called Tamura's features. There are six main features

based on characterizing elements of the texture based on the perception system. These are

- Contrast
- Directionality
- Coarseness
- Linelikeness
- Regularity
- Roughness

The details of each Tamura feature explain in the following sections. We have implemented and used all of them, and based on the results that we have derived, we have focused mainly on some of them.

3.2.2.1 Coarseness

Coarseness is one of the most important of Tamura's features. It is a fundamental textural feature that defines the amount of magnification of the pattern on texture. The coarser texture means, the bigger the elements of the patterns or the less repetition of the patterns on the texture's unit area. It has a direct relationship with the repetition of the elements of the image. The reverse of the coarse is fine, which means more pattern distribution with a smaller size. In short, on the same unit of window smaller count of the texture patterns means a coarser texture.

Using Tamura et al. approach, the six averages of the pixels with windows sizes 0,1,2,3,4,5 have to be evaluated around every pixel pi in the image to evaluate the material's coarseness.

It is possible to describe the above operation in a mathematical equation as follow [2]

$$f_{crs} = \frac{1}{n^2} \sum_{i}^{n} \sum_{j}^{m} 2^k p(i,j) \tag{3.2.3}$$

where $n \times m$ denotes the size of the image, and the sum computed for every pixel $p(i, j)$. k is the value that maximizes the difference between moving averages taken over $2^k \times 2^k$ neighborhood along with horizontal and vertical directions.

3.2.2.2 Contrast

The contrast feature of the image measures the variance of the grey level values in the image. It refers to the intensity difference between neighboring pixel values [2]. High contrast means the image has large differences in the intensity values of the neighboring pixels. According to [1] there are some factors other than gray-level distribution that have effects on contrast. These are

- Dynamic range of gray levels
- Polarization of the distribution of black and white on the gray-level histogram or ratio of black and white areas
- Sharpness of edges
- Period of repeating patterns

It is possible to describe the above operation in a mathematical equation as follow [2]

$$f_{con} = \frac{\sigma}{\left(\frac{\mu_4}{\sigma^4}\right)^{\frac{1}{4}}} \tag{3.2.4}$$

where σ is the standard deviation, and μ_4 is the fourth momentum of the image.

3.2.2.3 Directionality

Directionality is the property that defines the placement rule of the texture primitives. Int directional texture, the texture primitives, are placed somehow ordered in a specific direction. The orientation of the primitives is mostly directed in the same direction [2].

$$f_{dir} = 1 - r \cdot n_p \sum_{p}^{n_p} \sum_{\phi \epsilon w_p} (\phi - \phi_p)^2 \cdot H_D(\phi) \tag{3.2.5}$$

In Equation 3.2.5, the H_D is the histogram of the local direction, ϕ is the quantized direction code, ϕ_p is the p-th peak position of the H_D histogram, r is the normilizing factor, n_p is the number of the peak values in H_D and w_p is the range between valleys of the p-th peak.

3.2.2.4 Linelikeness

Linelikeness defines the shape of the texture primitives. Linelike texture primitive means that it has shape something like a straight line or wave. Overall, the orientation and distribution of the primitives do not matter. Directional textures are often linelike textures. Linelikeness of the texture can be computed as follow [2].

$$f_{lin} = \frac{\sum_i^n \sum_j^m P_{Dd}(i,j) cos((i-j)\frac{2\pi}{n})}{\sum_i^n \sum_j^m P_{Dd}(i,j)} \tag{3.2.6}$$

where $P_{Dd}(i,j)$ is the direction co-occurrence matrix with a size of $n \times m$ and every point is in the region with a max distance of d.

3.2.3 Regularity

The regularity of the texture defines placement variations of the primitives on the texture. Regular texture means the identical (regular) placement of the same or similar primitives. It is a measure of how regular the primitives are placed. The irregular texture is the texture on which primitives place irregularly or with a random arrangement. The regularity feature of the texture can be computed as follow [2].

$$f_{reg} = 1 - r(\sigma_{crs} + \sigma_{con} + \sigma_{dir} + \sigma_{lin}) \tag{3.2.7}$$

where r is normalizing factor and $\sigma_{crs}, \sigma_{con}, \sigma_{dir}, \sigma_{lin}$ are standard deviation values of $f_{crs}, f_{con}, f_{dir}, f_{lin}$

3.2.3.1 Roughness

The roughness of the texture refers to a geometric variation on the surface of the texture. Rough texture means more variation on the surface smoothness. On the other hand, the texture's smoothness means texture with a smooth surface without angled primitives on the surface. The roughness can be computed by adding the coarseness and the contrast [2].

$$f_{rgh} = f_{crs} + f_{con} \tag{3.2.8}$$

where f_{crs} is value of the coarseness and f_{con} is the value of contrast.

3.2.4 GLCM Features

One of the most straightforward image analyses is texture analysis using GLCM features, which Haralick [4] defined. GLCM texture analysis is proven to be related to the mechanism that the human perception(visual) system perceives the textures [4]. It is one of the most widely used texture analysis methods in image analysis, especially segmentation. Many research proved that the features suggested by Haralick are advantageous in texture analysis over others. GLCM is the matrix that defines the co-occurrence of the given matrix entries in a predefined direction. The direction should be any direction inside of the matrix. Figure 3.3 illustrates an example of GLCM matrix.

Twenty-two features could be calculated from the GLCM matrix: five are more useful and mainly used than others. These are:

- GLCM contrast
- GLCM dissimilarity
- GLCM homogeneity
- GLCM energy
- GLCM entropy

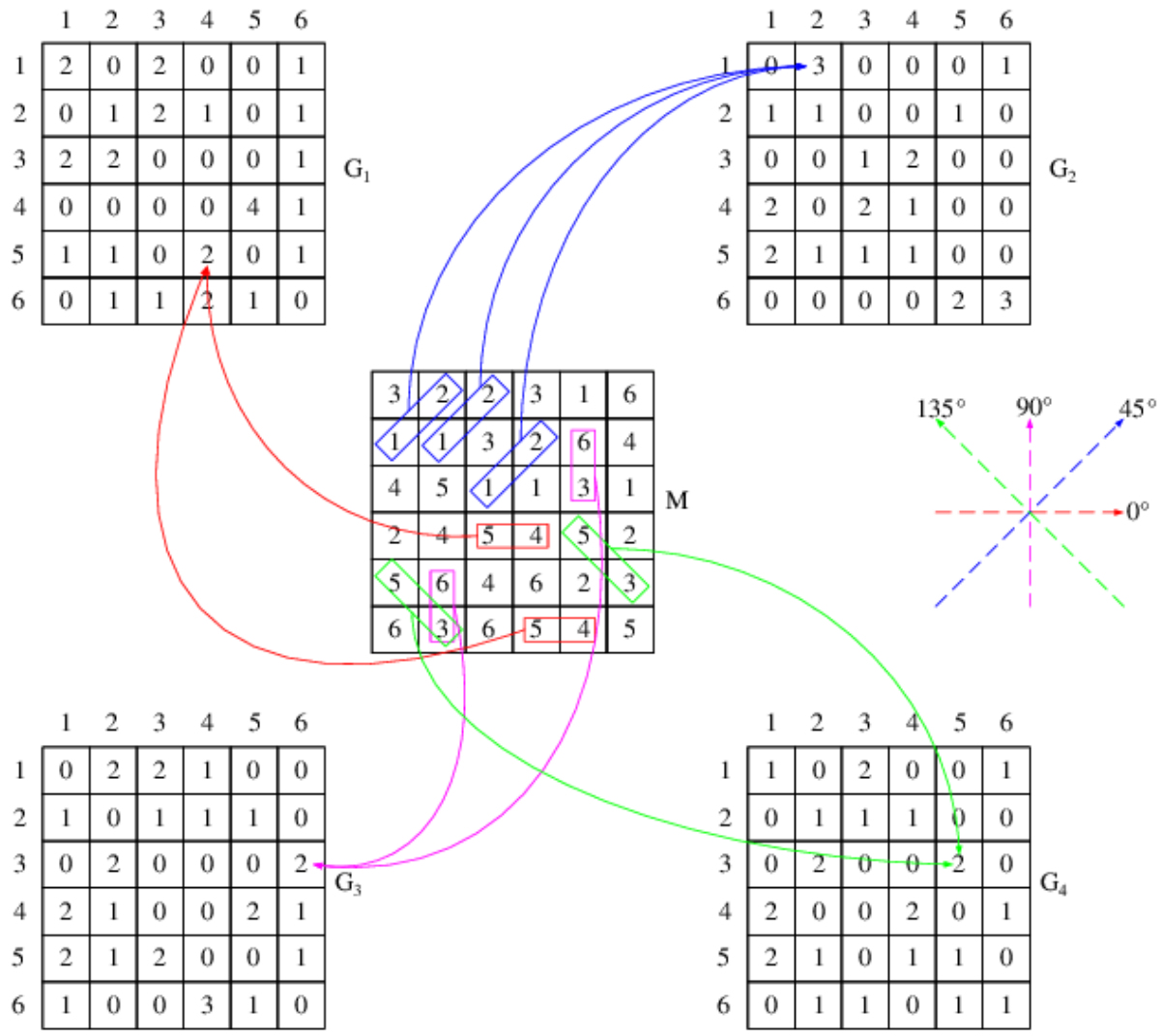


Figure 3.3: Example calculation of the GLCM matrix. $G1$ is a horizontal calculation of the GLCM matrix. $G2$ is a diagonal up calculation of the GLCM matrix. $G3$ is a vertical calculation of the GLCM matrix, and $G4$ is the diagonal down calculation of the GLCM matrix [8].

- GLCM correlation

3.2.4.1 GLCM Contrast

Contrast is the local variation of the gray level value in the co-occurrence matrix. Contrast is like linear dependency between neighboring pixels in grey level co-occurrence

matrix [4].

$$f_{cntrs} = \sum_{i,j}(i-j)^2 p(i,j) \tag{3.2.9}$$

In Equation 3.2.9, i and j are pixel coordinates or cell coordinates, and $p(i,j)$ is the value of the pixel or the cell in the matrix.

3.2.4.2 GLCM Dissimiliarity

Dissimilarity defines the amount or measure of the grey-level pairs' variation in the GLCM matrix. It is something like GLCM contrast. Unlike contrast, the GLCM dissimilarity grows quadratically [5].

$$f_{dsml} = \sum_{i,j}(i-j)p(i,j) \tag{3.2.10}$$

The contrast and the dissimilarity measure the same parameter using different weights. The value of the dissimilarity ranges between 0 and 1. It gets the maximum value when the neighboring pixels get the maximum values of the grey levels. In Equation 3.2.10, i and j are pixel coordinates or cell coordinates, and $p(i,j)$ is the value of the pixel or the cell in the matrix.

3.2.4.3 GLCM Homogenity

The GLCM homogeneity measures how uniforms are the pixel values or matrix entries are spread in the GLCM matrix [3]. The value of the image's homogeneity gets high when the pixels' values are the same or similar. When the pixel values changes in the GLCM direction are big, the GLCM homogeneity is getting low. Like dissimilarity, GLCM homogeneity takes values between 0 and 1. When there are no changes in the image or the ideal repetitive structural texture, it gets a value 1. If there is no repetitive texture or similarity in the image regions, it refers to an 'inhomogeneous image.'

$$f_{hmg} = \sum_{i,j}\frac{1}{1-(i-j)^2}p(i,j) \tag{3.2.11}$$

In Equation 3.2.11, i and j are pixel coordinates or cell coordinates, and $p(i, j)$ is the value of the pixel or the cell in the matrix.

3.2.4.4 GLCM Energy

GLCM energy is the measure of the local homogeneity in the image. The parameter measures the value of the uniformity of the texture [3, 5]. The value of the energy is related to the value with the homogeneity. It ranges between 0 and 1. The value of the energy of the solid image is 1.

$$f_{enrg} = \sum_{i,j} p(i, j)^2 \tag{3.2.12}$$

In Equation 3.2.12, i and j are pixel coordinates or cell coordinates, and p(i, j) is the value of the pixel or the cell in the matrix.

3.2.4.5 GLCM Entropy

The GLCM entropy defines the spatial disorder in the texture [3]. As it represents the disorder representation in the image, it is the opposite of the GLCM energy. The entropy's value gets high when the rate of the disorder in the texture is high, or there is no ordered pattern on the texture. If the image has a pattern or ordered texture, the entropy's value becomes a low value. It becomes zero on the single color image.

$$f_{entrpy} = -\sum_{i,j} p(i, j) \log p(i, j) \tag{3.2.13}$$

In Equation 3.2.13, i and j are pixel coordinates or cell coordinates, and $p(i, j)$ is the value of the pixel or the cell in the matrix.

3.2.4.6 GLCM Correlation

GLCM Correlation is the measure of the linear dependency of the grey levels of the neighboring pixels [7].

$$f_{crlt} = \frac{\sum_{i,j}(ij)p(i,j) - \mu_x\mu_y}{\sigma_x\sigma_y} \tag{3.2.14}$$

where 3.2.14 i and j are pixel coordinates or cell coordinates and p(i, j) is the value of the pixel or the cell in the matrix. μ_x, μ_y, σ_x and σ_y are the values of the means and the standard deviations of the pixels p_x and p_y.

3.2.5 Correlation Analysis

Correlation is the analysis of two sets of data to determine the degree of the linear or non-linear relationship between these data. There are different methods to do correlation analysis in data science. With this research's scope, we have used Spearman and Pearson Correlation analysis to get the correlation between the outputs of the feature analysis and class values. The correlation coefficient is the value that describes the correlation between the two sets. It ranges between -1 and 1. -1 value means a perfect negative correlation between two sets, and one means a positive correlation. The value 0 means there is no correlation or relationship between the distribution of the two datasets.

3.2.5.1 Pearson Correlation

Pearson correlation, also called Pearson's r correlation, measures the linear correlation between two datasets. Its value ranges between -1 and 1. The one means there is a linear correlation between the two sets. The -1 means there is a negative linear correlation. The value 0 indicates that there is no linear relationship or correlation

between the two sets. It is calculated as in Equation 3.2.15.

$$r_{xy} = \frac{n\sum x_i y_i - \sum x_i \sum y_i}{\sqrt{n\sum x_i^2 - (\sum x_i)^2}\sqrt{n\sum y_i^2 - (\sum y_i)^2}} \tag{3.2.15}$$

where n is number of observations, x_i is the value of x (for ith observation) and y_i is value of y (for ith observation) [54].

3.2.5.2 Spearman Correlation

The Spearman correlation, which is also called Spearman's ρ[56] correlation, is unlike Pearson's correlation, does not measure the value of the correlation between two datasets. It measures the strength and the direction of the monotonic association between two sets. It is a non-parametric measure of the rank correlation between these two parameters [57]. It is measured on a scale ranging from -1 to +1 as Pearson's correlation. The Spearman's rank correlation is calculated as in Equation 3.2.16.

$$\rho = 1 - \frac{6\sum d_i^2}{n(n^2-1)} \tag{3.2.16}$$

where d_i is difference between the two ranks of each observation, n is number of observations [55].

3.2.5.3 Logistic Regression

Regression analysis is a predictive analysis technique to analyze a relationship between one of the dependent variables and one or a series of independent variables. It is one of the most widely used tools to analyze multi-factor data [58]. The method is called multiple regression when there are multiple independent variables. There are two types of regression analysis: linear regression and logistic regression. Logistic regression is the regression analysis that is mainly used in data analysis. It calculates the probability of the event occurrence, which is a binary value.

3.3 Classification Using Deep Learning

Because DL is getting popular on classification and other tasks due to its outstanding performance, it improves and is easy to use through open-source tools. Most of the SOTA object classifiers are based on deep convolutional neural networks (DCNNs). Due to DL's performance on similar tasks, we have tested their performance for softness analysis and pilling assessment as a second method. Many network architectures exist in the literature. They require fine-tuning and transfer learning. DCNNs have many learnable parameters. They need large datasets on which to be trained and tested. Dataset generation is one of the most critical tasks in using DCNNs. In Chapter 4, details of the algorithms that we have used for the dataset generation have been described.

3.3.1 Development Environment

We have used PyTorch [59], and PyQt [60] as the main components of our development environment. PyTorch is a computing framework that enables a flexible environment for developers to develop and test deep learning-based computations. It is a Python [62] based framework that also includes many valuable packages to speed up the development procedure. The PyQt is the combination of Python and Qt, which makes it possible to develop GUI applications on supported platforms such as Windows, Linux, Mac OS [61].

We have used pre-trained networks, which have been included in the PyTorch library. There are many pre-trained models ready to use as a pre-trained or without using the pre-training feature. We mainly used pre-trained models applied fine-tuning using our dataset. We experimented with several models such as *mobilenetV2* [70], *resnet18* [72], *densenet121* [64], and *densenet161* [64].

CHAPTER 4

METHODOLOGY

4.1 Dataset Collection

Over the course of this study, various fabric images have been collected to serve as datasets for automated softness and pilling assessment. Initially, some sample fabrics were sent by Arcelik to our side, and we captured their images using a Canon EOS 550D DSLR camera. These images were captured under near fluorescent illumination. A sample result is shown on the left part of Figure 4.1. As can be seen in this figure, the camera viewed the sample at an angle to prevent casting a shadow on the sample with respect to the illuminant, which was placed over the sample. This, however, caused some problems as the details on the top part of the fabric were blurred due to being more distant and having been captured from a narrower angle. It was also difficult to control the positioning of the camera and the samples and there were some variations between the images despite a tripod was used.

We also received several batches of the digital images captured by the Arcelik team for softness assessment. An example is shown on the right of Figure 4.1. The first set of examples was not very useful because of the lower resolution of the examples. The sample identifier label in the top-left and the capture date in the bottom-right corners further reduced the usable sample area.

Despite these problems, the initial datasets were useful for our first analyses. We then decided to capture a new set of images under more controlled conditions. To this end, the Arcelik team prepared a controlled capture environment as shown in Figure 4.2. After further discussions with the Arcelik team, we decided to generate our final data by using a new setup in which a higher resolution smartphone replaced the camera.

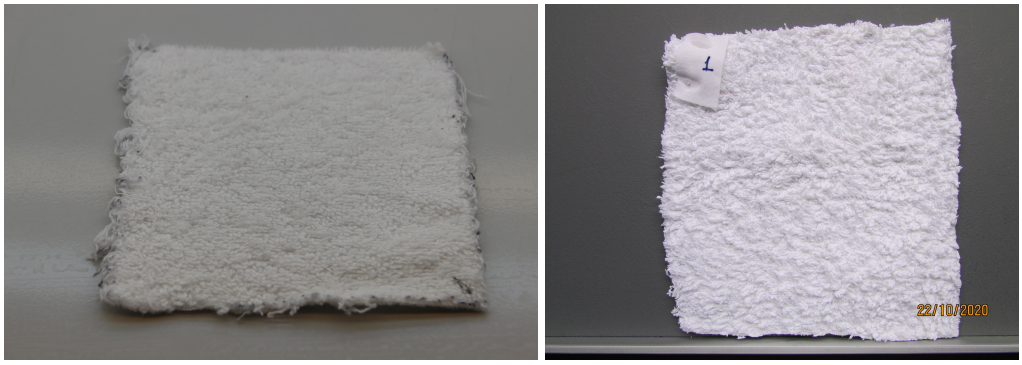


Figure 4.1: Example fabric samples. The left image was captured by a Canon EOS 550D DSLR camera on our side. The right image was sent by the Arcelik team.

This setup is shown in Figure 4.3. The smartphone that was used was a Mi8-Lite, which has a 12MP camera with a sensor of 1/2.55 inch. The light source in the setup environment was controlled by using an artificial lightbox (artificial daylight / 6500K, 20W). The properties of the samples were pre-defined, which are listed in Table 4.1. Each sample was 20x20 cm in dimension and was an optic white-colored piece of towel. As can be seen in the top-left corner of the towel in Figure 4.2, each sample was marked with an expert jury score.

Table 4.1: Properties of samples (towels).

Ground warp yarn number (Ne)	20/2
Pile warp yarn number (Ne)	16/1
Weft warp yarn number (Ne)	16/1
Weight of a towel (gsm)	450
Color	Optic white
Sample dimension (cm)	20x20

To capture sample images for the pilling analysis, we have used the same setup with the Mi8-Lite smartphone as the imager. In Figure 4.4, the final environment of the pilling sample capture is visualized. The light was provided from the top of the samples, which made the visualization of the pilling artifacts more distinguishable. The samples were white socks with different amounts of pilling on the surface for the pilling analysis. The socks were pilled under a special environment in which ISO

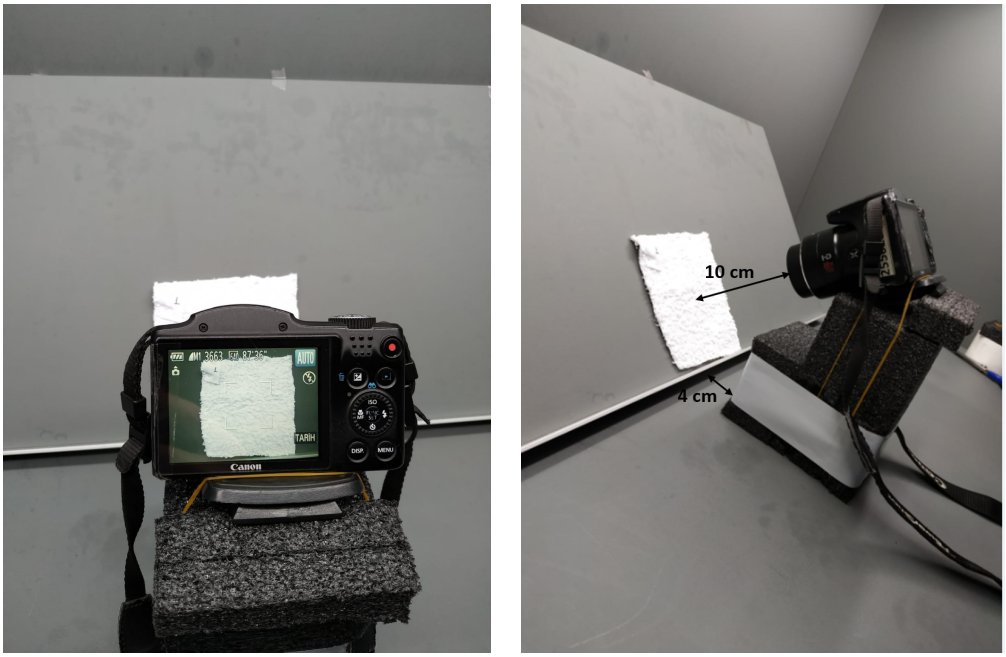


Figure 4.2: Initial camera setup of the Arcelik data.

Figure 4.3: The final setup to capture sample images that are ready to use in softness analysis.

12945-2:2000 test standard for the textiles was considered. All the samples were pilled in a laboratory environment using a Martindale test machine [36, 37]. The

Figure 4.4: Pilling sample capture final setup.

definitions of the pilling scores are listed in Table 4.2.

To summarize, the final datasets that were used in this study were captured by the Arcelik team using the methodology described above. Originally the datasets contained expert scores ranging from 1 to 5 in 0.5 increments giving rise to a total of 9 classes. However, the preliminary analysis showed that a 9 class classification problem with relatively small training data and minute differences between adjacent class samples is extremely difficult. As a result, we decided to reduce the class size to 5 for both datasets. We achieved this using the techniques described below.

The original softness assessment dataset contained 154 sample images distributed over 9 classes (from 1 to 5 in half-integer increments). In reality, there were 77 fabrics, but they were captured from two faces yielding 154 images. If a fabric was given an integer score, we placed both the fabric's front and back face images in the corresponding class. On the other hand, if a fabric was given a half-integer score, its front face and back face were placed in separate integer classes that are nearest to the actual score. The resolution of each image was 3024×4032.

The pilling dataset contained 291 images again scored between 1 to 5 in half-integer

Table 4.2: Definition of the pilling degrees according to ISO 12945-2:2000 [31].

Pilling degree	Description	Detail according to ISO-12945-2:2000[66]
1	Very severe pilling	Severe pilling covering whole of the fabric surface.
2	Severe pilling	Distinct surface pilling. Pills of various size and density covering a large proportion of the surface.
3	Moderate pilling	Moderate surface pilling. Pills of varying size and density partially covering the surface.
4	Low pilling	Slight surface fuzzing.
5	No pilling	No change in fabric aspect.

Class	Softness Dataset	Pilling Dataset
1	33	14
2	29	63
3	31	60
4	22	104
5	39	50

Table 4.3: Distribution of samples across classes for the softness and pilling datasets.

units. Different from the softness assessment dataset, the same fabric was not captured from the two faces. Therefore, to reduce the class count to 5 we used a different approach. For each fabric marked with a half-integer score, we assigned it to either the lower or upper integer class. This way 50% of such fabrics marked with an in-between score were assigned to the lower group and the other 50% to the higher group. The resolution of the images was the same as for the softness images. Table 4.3 shows the number of images in each class. As can be seen from the table, the number of samples in each class was not equal. This difference was especially more pronounced for the pilling dataset. Precautions were taken to address this problem, as explained in the next section.

4.2 Analysis

In this section, we describe the analysis methodology adopted in this thesis. The same techniques are used for the softness and pilling assessment tasks due to the similarity between the tasks. During the analysis, we created 224×224 sized patches from the original photographs. These patches are created by using a sliding window technique without overlaps. The reason for selecting this resolution was that all of the deep learning models we used expected this resolution. It also allowed us to create multiple samples from a single photograph, thus extending our sample size for hand-crafted feature analysis as well.

4.2.1 Handcrafted Feature Analysis

Our analysis began with the handcrafted features. To this end, we first computed all of the handcrafted feature values for the patches extracted from the samples. The results were collected in a CSV file for further analysis. In the analysis, we first computed the Spearman correlations with the individual features. This helped identify the most promising correlations for further analysis and leave out those that do not appear to have a significant positive or negative correlation[1]. We decided to use the Spearman correlation instead of Pearson correlation as the former is directly related to ranking

[1] Negative correlations were equally useful as the positive correlations as for them the negated feature value would be positive.

while the latter is more suitable for rating judgments. But overall, there was a high degree of correlation between both measures.

Based on this initial analysis, we selected the features whose absolute value of the correlations was at least 0.6. These features are used in the following classification analysis. For this purpose, we experimented with various classifiers that were readily available in the *scikit-learn* machine learning library for Python [75]. In particular, we used linear regression, multinomial logistic regression, ordinal logistic regression, and support vector machines. For reporting the results, we used the mean absolute error, overall accuracy scores, and per-class recall, precision, and F1 scores as the success measure of the classifiers.

4.2.2 Deep Learning Based Analysis

For DL analysis, we used the deep learning models that are implemented in the PyTorch library [76]. There are various models in this library such as the AlexNet [65], ResNet [72], DenseNet [64], VGG [67], SqueezeNet [68], ShuffleNet [69], MobileNet [70], GoogleNet [71], and Inception [66]. Each model also comes with several variants such as DensetNet121 [64], DenseNet161 [64], and DenseNet169 [64]. These variations correspond to the same underlying base model's configurations (i.e. the number of filters in different layers). As a result, there were about 20 candidate models with which we could experiment. We conducted a preliminary analysis to choose the most suitable models based on the model's success and whether they could run on our test computer. Some models required more GPU memory than was available. Based on this preliminary analysis, we found the DenseNet models to be the most suitable. Among the DenseNet models, the DenseNet121 model, which is the simplest among them, appeared to work well. We, therefore, focused on this model in our experiments. The architecture of this model is shown in Figure 4.5. The 121 in the naming of this model is computed by $121 = 5+2(6+12+24+16)$ where 5 indicates the total number of convolution-pooling (1), transition (3), and classification (1) layers. Each dense block successively contains 6, 12, 24, and 16 filters. Because dense blocks are made up of two convolutional layers, their sum is multiplied by two.

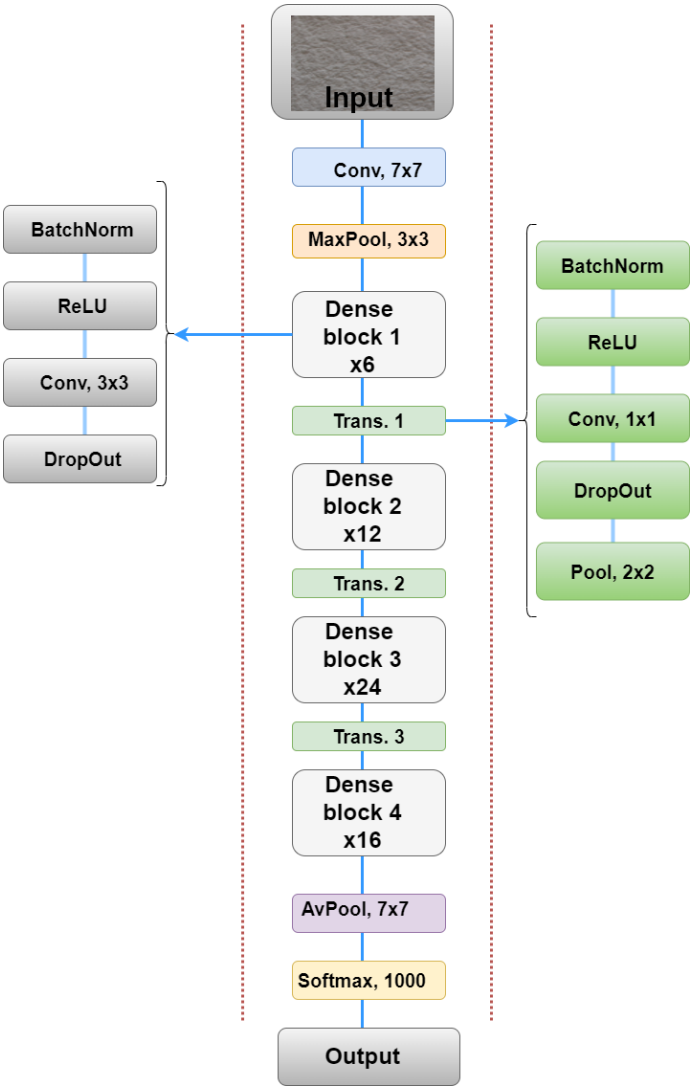


Figure 4.5: Architecture overview of DenseNet121.

4.2.3 Cross Validation

In order to test the success of both HCF and DL approaches, we used K-fold cross-validation. In this type of validation, the dataset is divided into K non-overlapping subsets. The models are developed from the $K-1$ parts, and they are tested on the part that is left out. This process is repeated K times, and the results are averaged to compute the final scores. Because there was an imbalance in the class sizes, we used stratified K-fold cross-validation, which performs the partition according to the number of elements in each class. For HCF analysis, each fold was divided into two

parts, one for model building (80%)and the other for testing (10%)[2]. For DL analysis, each fold was subdivided into three parts that correspond to training (80%), validation (10%), and testing (10%) subsets. It was ensured that training, testing, and validation images were always selected from different source photographs to represent a more realistic real-world evaluation scenario. While the initial tests were performed on our own computers, the final results reported in this paper were performed at TUBITAK ULAKBIM, High Performance and Grid Computing Center (TRUBA resources).

4.2.4 Data Augmentation

In order to increase the training sample size, we used data augmentation for deep learning approaches. During the training each patch was given as:

- Original
- Randomly flipped vertically
- Randomly flipped horizontally
- Randomly color jittered to alter brightness, contrast, saturation, and hue

The transformations are applied independently of each other. This means that a training patch could undergo vertical flipping, horizontal flipping, as well as color jittering. We experimented with 10% to 20% color jittering amounts. The flipping probability in each direction was 50%. As explained in the following subsection, we observed the data augmentation process to help improve accuracy and recall scores.

[2] 10% of the data, which corresponds to validation samples for DL, was not used for a fair comparison between HCF and DL based classification approaches.

CHAPTER 5

RESULTS AND EVALUATION

5.1 Hand Crafted Features

5.1.1 Feature Correlations

5.1.1.1 Softness Assessment Task

As our first set of results, we share the correlations of the individual hand crafted features with the expert scores for the softness assessment task. Figure 5.1 shows the correlation scores using both Spearman and Pearson correlation coefficients. In both graphs, the x axis represents the feature types and the y axis represents the value of the correlation coefficients. Some immediate conclusions can be drawn from these graphs. In the following we use the Spearman correlation scores but because both graphs are mostly similar, the following comments can be made for both of them.

Firstly, some feature types have very low correlation scores. These are standard deviation (0.01), Tamura directionality (0.11), Tamura line-likeness (0.06), Tamura regularity (-0.14) and GLCM correlation (0.13). It can be said that these features are not very useful for assessing the softness of a fabric. On the opposite side of the scale, some features have relatively high correlation. The features with ≥ 0.6 Spearman correlation are wavelet energy (0.68), GLCM contrasts (0.69), GLCM homogeneity (0.69), and GLCM energy (0.62). These features are promising and it can be expected that their combination can lead to a good classifier for softness assessment.

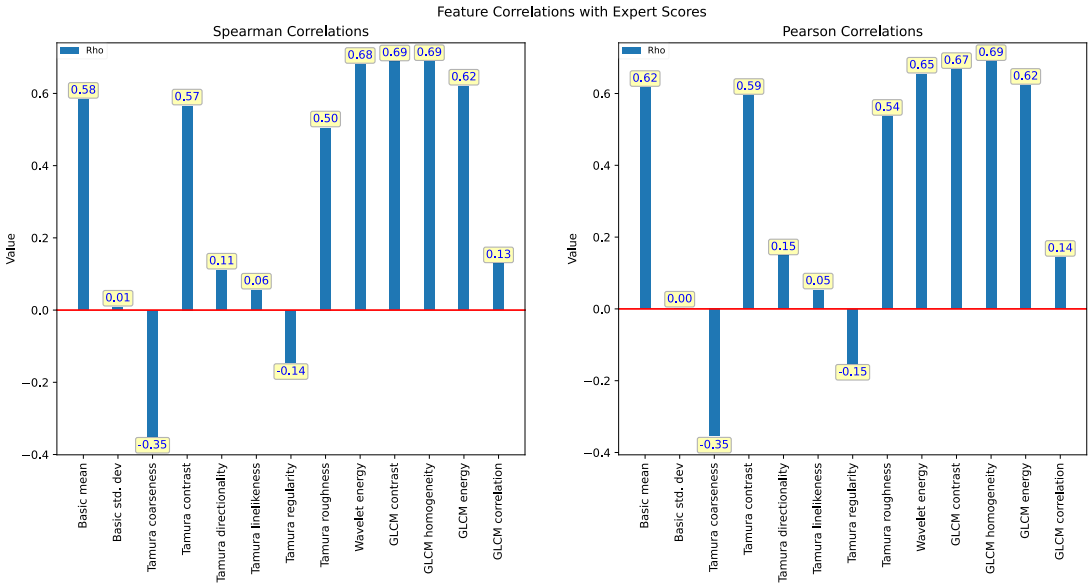


Figure 5.1: Feature correlation results for softness assessment.

5.1.1.2 Pilling Assessment Task

We used the same methodology as we used above for the pilling assessment task as well: we computed the Spearman and Pearson correlations for all of the 13 textural features. The results are shown in Figure 5.2. However, different from the softness assessment, none of the features appeared to correlate well with the expert scores. Note that negative correlations are not important; their negation could be used to convert them to positive correlations. However, the absolute value of all features were below 0.5 indicating low correlation. This led us to conclude that the low level textural features that we used in this study are not suitable for assessing the pilling degree of fabrics. As a result, we did not try to train hand crafted feature based classifiers as their performance can also be expected to be low.

5.1.2 Classification

5.1.2.1 Softness Assessment Task

The problem of classification is to determine a class label from a number of feature representations. As discussed in the previous section, we experimented with several

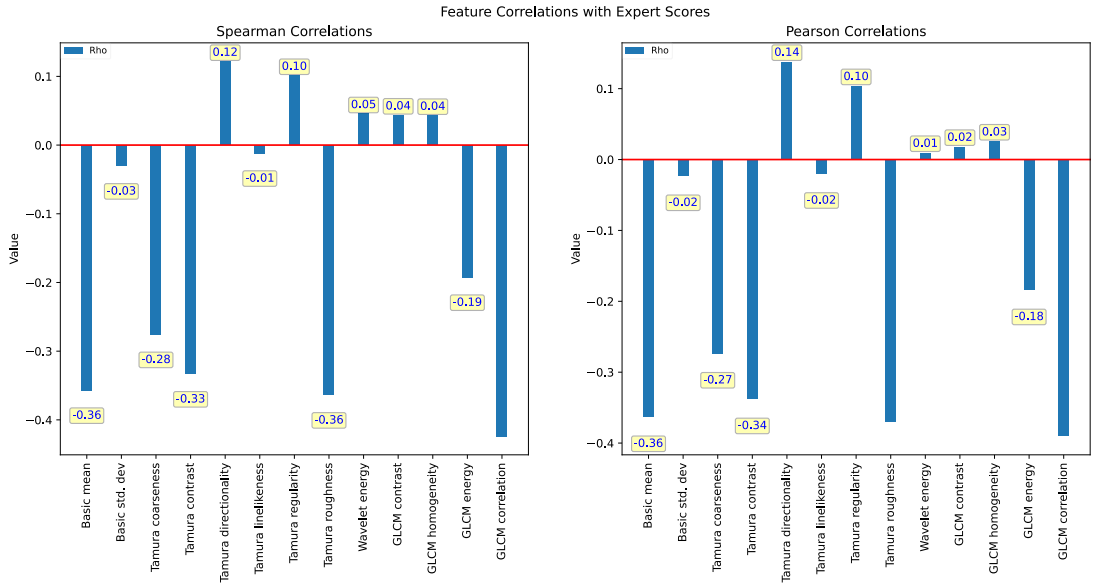


Figure 5.2: Feature correlation results for pilling assessment.

classifiers namely linear regression, multinomial logistic regression, ordinal logistic regression, and support vector machines. All of these classifiers are trained using stratified cross validation by splitting the data into 10 folds according to the distribution of samples in each class. For each fold, mean absolute error (MAE) is computed which is defined as:

$$\text{MAE}(y, \hat{y}) = \frac{1}{n} \sum_{i=1}^{n} |y_i - \hat{y}_i|, \tag{5.1.1}$$

where n is the number of test samples for that fold. As the overall error result, per-fold MAEs are averaged to obtain a single error score for each of the classifiers. The results are reported in Table 5.1. According to this table, it can be seen that the ordinal logistic regression introduces the least amount of error. Furthermore, the error rate of the classifiers are all around 1, which indicates that while the models are introducing errors, the average error rate is not very high. In other words, while the models may frequently mispredict a class whose original label is 2 as 1 or 3, mispredicting it as 5 is much rarer.

This issue can be better understood by inspecting the confusion matrix for each classifier. A confusion matrix is a compact representation of the results of a classification experiment. The i^{th} row and the j^{th} column of this matrix contains the number of samples whose true label is i and the predicted label is j. The more diagonally dominant

Classifier	Mean Absolute Error (averaged over folds)
Linear regression	1.02
Multinomial logistic regression	1.10
Ordinal logistic regression	0.98
Support vector machine	1.08

Table 5.1: The mean absolute error results for different classifiers on the softness assessment task.

a confusion matrix is, the more accurate is the corresponding classifier. In this study, we accumulated the confusion matrices of each fold to obtain an overall confusion matrix that represents the results of the entire experiment. For display purposes, the entries of the confusion matrix are normalized. This is achieved by dividing every row by the sum of the elements in that row. The resulting confusion matrices for each model are depicted in Figure 5.3. It can be seen that most classifiers are diagonally dominant: larger misclassification errors are rarer compared to smaller misclassification errors.

Several error metrics can directly be computed from the elements of the confusion matrix such as accuracy, precision, recall, and F1 score. The overall accuracy is defined as:

$$\text{acc}(y, \hat{y}) = \frac{1}{n} \sum_{i=1}^{n} \mathbb{1}(y_i == \hat{y}_i), \tag{5.1.2}$$

where $\mathbb{1}(x)$ is the indicator function. This accuracy value can be computed by dividing the sum of the diagonal elements to the sum of the all elements of the confusion matrix. The resulting accuracy values for each classifier are given in Table 5.2. According to this, the multinomial logistic regression and SVM classifiers perform better than the ordinal logistic regression and linear classifiers. It is also interesting to note that the multinomial and SVM classifiers have higher accuracy than the ordinal one despite the latter having a smaller mean absolute error. We suspect that this is because of the nature of the accuracy metric; smaller and larger errors contribute equally to the result.

Finally, we report the precision, recall, and F1 scores for each classifier. These are

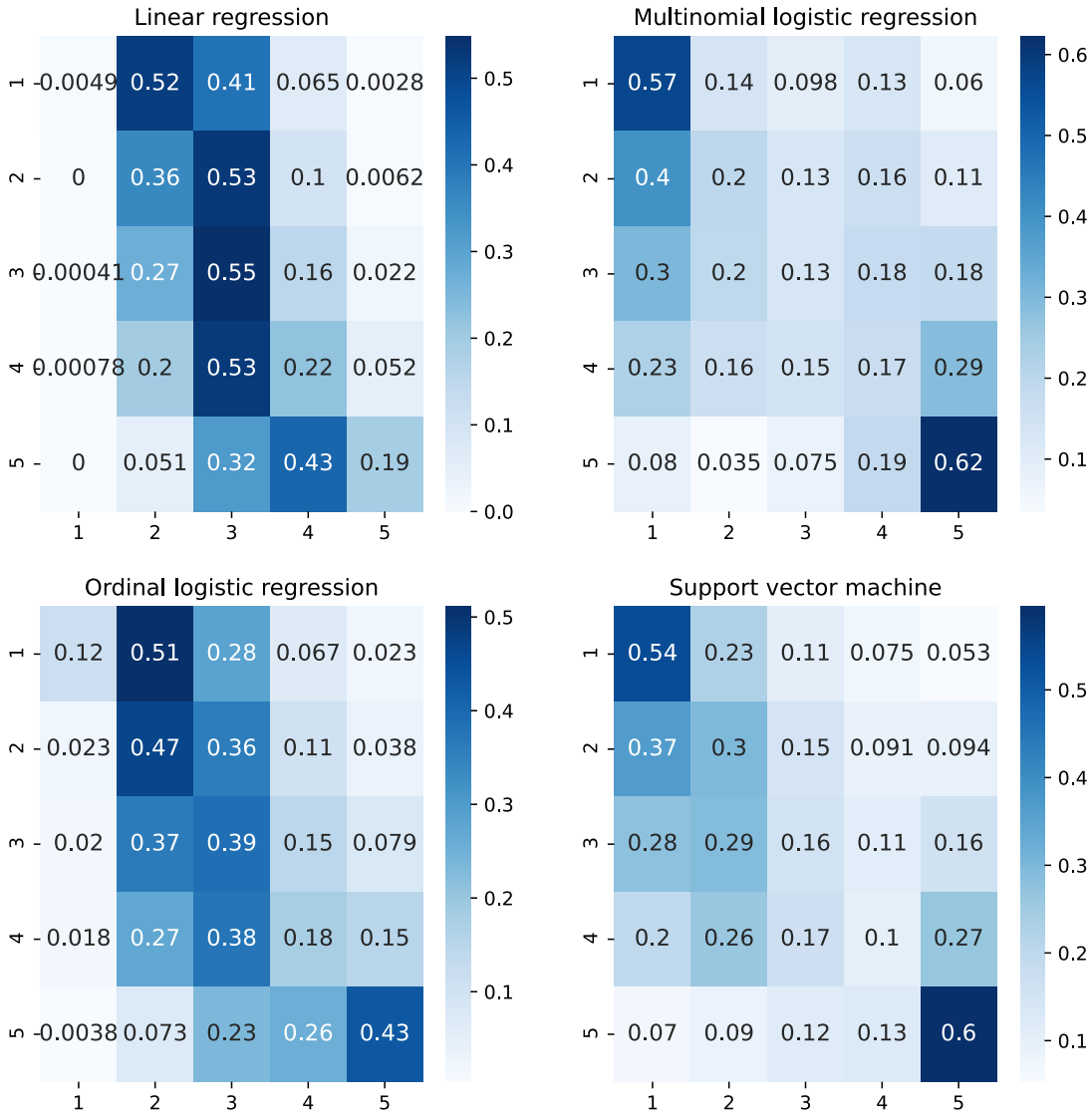


Figure 5.3: The aggregated (over folds) confusion matrix for each classifier for the softness assessment task. The confusion matrix is normalized in each row.

Classifier	Accuracy
Linear regression	0.26
Multinomial logistic regression	0.37
Ordinal logistic regression	0.33
Support vector machine	0.37

Table 5.2: Overall accuracy of classifiers for the softness assessment task.

arguably more revealing measures than the overall accuracy as they give us a chance to observe the performance of a classifier for each class. Precision is defined as the ratio of true positives to sum of true and false positives. It is defined as:

$$\text{pre} = \frac{\text{TP}}{\text{TP} + \text{FP}}, \tag{5.1.3}$$

where TP and FP represent the number of true positive and false positive predictions. Recall is defined as the ratio of true positives to all relevant items of a given class. It is defined as:

$$\text{rec} = \frac{\text{TP}}{\text{TP} + \text{FN}} \tag{5.1.4}$$

where FN represents the number of false negative predictions. Both precision and recall can be computed from the confusion matrix. The precision value for each class can be computed by dividing the diagonal element in each row (TP) to the sum of the all elements in that column (TP + FP). The recall value for each class can be computed by dividing the diagonal element in each row (TP) to sum of all elements in that row (TP + FN). Finally, the F1 score represents the harmonic mean of precision and recall. It is defined as:

$$\text{F1} = 2\frac{\text{pre} \times \text{rec}}{\text{pre} + \text{rec}} \tag{5.1.5}$$

These metric values of each classifier are listed in Tables 5.3, 5.4, and 5.5. According to these tables and the confusion matrices shown in Figure 5.3 it can be said that multinomial and SVM classifiers do a good job in classifying the first and the fifth classes. However, they do not perform well for classifying the medium class values. The second class is best classified by the ordinal and the third class is best classified by the linear regressors. In general, the fourth class appears to present a challenge for all classifiers.

5.2 Deep Learning

5.2.1 Softness Assessment Task

As explained in the previous section, we used the DensetNet121 model for classification of fabric samples into softness classes. Deep learning models can be trained from scratch but experiments have shown that models whose weights have already

Classifier	Precision (1, 2, 3, 4, 5)
Linear regression	$0.84, 0.25, 0.24, 0.15, 0.78$
Multinomial logistic regression	$0.39, 0.27, 0.24, 0.15, 0.59$
Ordinal logistic regression	$0.67, 0.27, 0.24, 0.17, 0.69$
Support vector machine	$0.40, 0.25, 0.23, 0.14, 0.60$

Table 5.3: Per-class precision rates of the HCF classifiers for the softness assessment task.

Classifier	Recall (1, 2, 3, 4, 5)
Linear regression	$0.00, 0.36, 0.55, 0.22, 0.19$
Multinomial logistic regression	$0.57, 0.20, 0.13, 0.17, 0.62$
Ordinal logistic regression	$0.12, 0.47, 0.39, 0.18, 0.43$
Support vector machine	$0.54, 0.30, 0.16, 0.10, 0.60$

Table 5.4: Per-class recall rates of the HCF classifiers for the softness assessment task.

Classifier	F1 Score (1, 2, 3, 4, 5)
Linear regression	$0.01, 0.29, 0.34, 0.18, 0.31$
Multinomial logistic regression	$0.47, 0.23, 0.17, 0.16, 0.60$
Ordinal logistic regression	$0.20, 0.34, 0.30, 0.17, 0.53$
Support vector machine	$0.46, 0.27, 0.19, 0.12, 0.60$

Table 5.5: Per-class F1 scores of the HCF classifiers for the softness assessment task.

been computed over a large scale dataset can provide better performance for new classification tasks. To this end, we used the DenseNet121 architecture pretrained on ImageNet [73] and finetuned it using the dataset of the current study. We also experimented with training the entire network from scratch, but starting from a pretrained architecture noticeably improved the classification scores.

By default, the pretrained DenseNet121 architecture outputs 1000 classes corresponding to the number of object categories in the ImageNet dataset. For finetuning, we replaced this last layer to output 5 classes. Furthermore, we wanted to design our loss function in an ordinal sense. In other words, we wanted to create a loss function which penalizes larger class errors more (e.g. 1 is classified as 5) compared to smaller class errors (1 is classified as 2). We therefore applied a sigmoid activation function on the outputs of the final 5 neurons. This serves the purpose of converting negative and positive absolute values into belief values in range $[0, 1]$. The sigmoid activation function is defined as follows:

$$f(x) = \frac{1}{1 + \exp(-x)}. \tag{5.2.1}$$

To perform ordinal regression, we also converted the target labels into one-hot-like encodings. This conversion was defined by the following mappings:

- Class 1: [1, 0, 0, 0, 0]
- Class 2: [1, 1, 0, 0, 0]
- Class 3: [1, 1, 1, 0, 0]
- Class 4: [1, 1, 1, 1, 0]
- Class 5: [1, 1, 1, 1, 1]

The loss function was then defined as the weighted mean squared error between the network outputs and the class representations using the above mappings:

$$\text{loss}(y, \hat{y}) = \frac{1}{n} \sum_{i=1}^{n} w_i (y_i - \hat{y}_i)^2, \tag{5.2.2}$$

where n is the number of classes, w_i is the weight of each class, y is correct label encoded as above, and $\hat{y}$ are the sigmoid applied network outputs. The weights were

chosen as inversely proportional to the number of samples we had for each class. This allows undersampled classes to also have more influence on the loss function. In particular, the weight vector was defined as:

$$w = [\frac{100}{33}, \frac{100}{29}, \frac{100}{31}, \frac{100}{22}, \frac{100}{39}] \tag{5.2.3}$$

Note that this loss function pushes the network outputs to resemble the one-hot-like encodings described above. For instance for a sample that belongs to the 3^{rd} class, after a sufficient amount of training, the network starts to output values like $\hat{y} = [0.99, 0.98, 0.97, 0, 0]$. We can then convert these encodings back to class labels by counting the number of elements from left-to-right whose value is above 0.5 (we stop counting as soon as we find an element whose value is less than 0.5). In the above example, we can deduce that the predicted label of $\hat{y}$ is 3 because there are three values with > 0.5 value from left-to-right. For the updates of network parameters, we used Adam optimizer with a learning rate of 0.0005.

Other than these details, we performed a standard network training process. The dataset was divided into training, validation, and test sets as explained in Chapter 4. We experimented with using techniques such as drop-out, weight regularization, and data augmentation to improve the quality of training. Our observation, however, has been that softness assessment is a challenge task for deep networks as we could not obtain a set of hyper-parameters that consistently reduces validation loss over the epochs of training. An example to illustrate this point is shown in Figure 5.4 for one of folds during cross validation. Here, it can be seen that while the training loss drops and training recall increases reliably (blue lines), validation loss and recall has a more haphazard behavior (orange lines). They change as desired initially but start to fluctuate starting with the fifth epoch. This is generally an indicator of memorization: the network becomes tuned to the training data and does not perform well on unseen validation data. Using the techniques such as drop-out, weight regularization, and data augmentation, this problem was somewhat mitigated but could not be entirely avoided.

The results of our experiments using different sets of hyper-parameters values are summarized in Table 5.6. According to this table, the best overall image based accuracy was obtained using the set of parameters shown in the third row of this table.

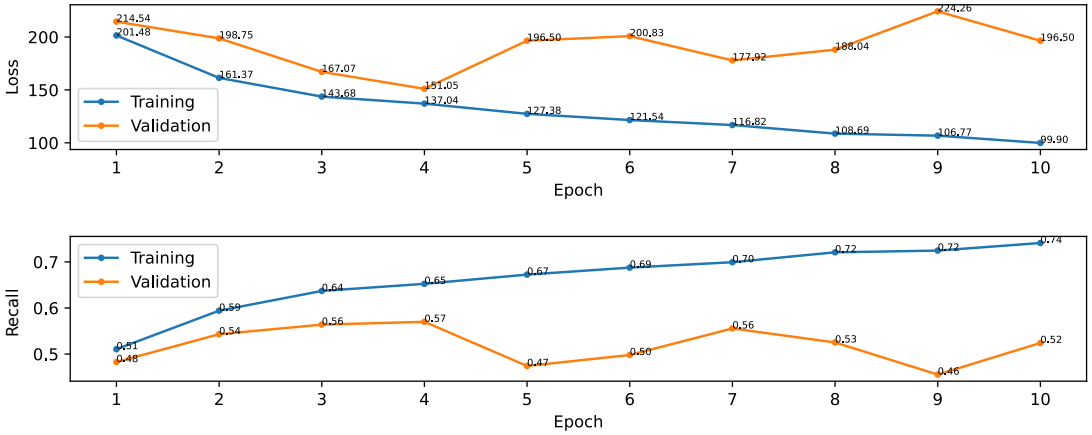


Figure 5.4: Change of mean squared error and average recall rates during one fold of the training.

Hyper-parameters	Accuracy
Drop-rate $= 0$, weight-regularization $= 0$, color-jitter $= 0$	0.53
Drop-rate $= 0$, weight-regularization $= 10^{-4}$, color-jitter $= 0.1$	0.52
Drop-rate $= 0$, weight-regularization $= 10^{-4}$, color-jitter $= 0.2$	$\mathbf{0.55}$
Drop-rate $= 0.1$, weight-regularization $= 10^{-4}$, color-jitter $= 0.2$	0.49

Table 5.6: Image based accuracy (see text for details) results for a different combination of hyper-parameters. The best result is shown in the third row.

The definition of image based accuracy is given below.

Overall, we observed that the trained networks appeared to perform considerably well compared to the HCF classifiers. In Table 5.7 we report the same precision, recall, and F1 score metrics as was done for HCFs. For instance, if we compare the F1 score row of this table with the corresponding best score for HCFs (multinomial regression row in Table 5.5), we can see that the DL results are consistently better for each class.

As the performance of DL based classifier was higher, we decided to perform an extra experiment in which we combined the results of all patches for a single test fabric to produce a single score for that fabric. As the resolution of each test image and patch were 3024×4032 and 224×224 respectively, we obtained 234 patches for each test image. Each patch had a classification score and we combined them using

Metric	Per-class value (1, 2, 3, 4, 5)
Precision	$0.59, 0.33, 0.35, 0.28, 0.74$
Recall	$0.58, 0.42, 0.35, 0.19, 0.74$
F1 Score	$0.58, 0.37, 0.35, 0.23, 0.74$

Table 5.7: Precision, recall, and F1 score results for deep learning based classification on the softness assessment task.

Metric	Per-class value (1, 2, 3, 4, 5)
Precision	$0.67, 0.31, 0.40, 0.24, 0.78$
Recall	$0.48, 0.45, 0.45, 0.18, 0.72$
F1 Score	$0.56, 0.37, 0.42, 0.21, 0.75$

Table 5.8: Precision, recall, and F1 score results for deep learning based classification on the softness assessment task using the mean patch results for each image.

two different approaches. In the first approach, we used the rounded mean value of the classification scores. In the second approach, we used the majority voting result of the patches. The results of both image-based approaches are shown in Tables 5.8 and 5.9. As can be seen from these tables, the majority voting results appear to be produce better metric scores compared to the mean approach. This can be expected as majority voting is more robust against outlier predictions.

An interesting observation can be made by comparing the metric values in Table 5.9 with the number of fabric samples that we had in our dataset (see Table 4.3). The

Metric	Per-class value (1, 2, 3, 4, 5)
Precision	$0.69, 0.37, 0.41, 0.38, 0.76$
Recall	$0.67, 0.45, 0.42, 0.23, 0.82$
F1 Score	$0.68, 0.41, 0.41, 0.29, 0.79$

Table 5.9: Precision, recall, and F1 score results for deep learning based classification on the softness assessment task using the majority voting of the patch results for each image.

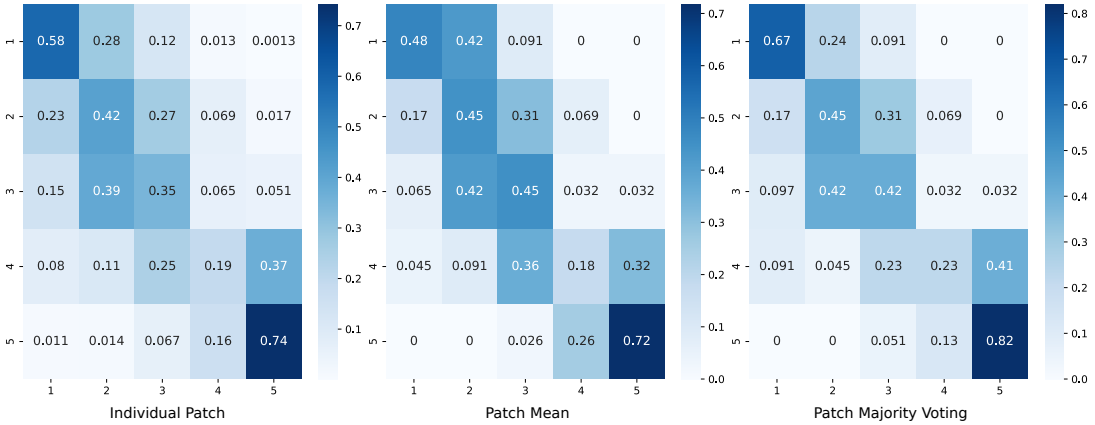


Figure 5.5: Confusion matrices for individual patch, patch mean, and patch majority voting results for the softness assessment task.

scores are almost perfectly correlated with the number of samples present in each class. For instance, the best scores are obtained for the fifth class, which also contains the highest number of samples (39). On the other end, the worst scores are obtained for the fourth class, which also contains the fewest number of samples (22). This is despite the fact that we weighted the loss function inversely proportionally to the number of samples using Equation 5.2.3. This suggests that if we had more and equal number of training samples for each class, the results could have been significantly improved.

To give a visual summary of the results, we share the confusion matrices for individual patch, patch mean, and patch majority voting confusion matrices in Figure 5.5. It can be seen that the matrices are predominantly diagonal suggesting that the predictions are generally accurate. Also the mispredictions are generally made into neighboring classes suggesting low errors. For example, if we focus on the fourth row of the majority voting matrix on the right of this figure, we can see that class 4 is mispredicted as class 3 23% of the time and as class 5 41% of the time. Mispredictions into more distant classes such as 1 and 2 are much rarer.

Finally, we share the overall accuracy results for the patch-based, mean-based, and majority-voting-based approaches. The overall accuracy, which was defined in Equation 5.1.2, can be intuitively understood as the sum of the diagonal elements in the confusion matrix to the sum of all elements in the matrix. In other words, this num-

Approach	Accuracy
Individual patch	0.49
Mean patch per image	0.49
Majority vote patch per image	0.55

Table 5.10: Overall accuracy of different prediction approaches for softness assessment using deep learning.

ber represents the ratio of exact matches to all predictions. The results are reported in Table 5.10. Given that the chance value would be 0.20 (random guessing among 5 classes) and the best HCF classifier had an accuracy of 0.37 (see Table 5.2), we can conclude that deep learning based classification has a plausible performance.

5.2.2 Pilling Assessment Task

To perform pilling degree classification using deep learning, we used the same network architecture as we did for the softness assessment task. We also used the mean squared error loss function given in Equation 5.2.2, but the weights were updated in accordance with the number of samples in each class. Specifically, we used the following weight vector:

$$w = [\frac{100}{14}, \frac{100}{63}, \frac{100}{60}, \frac{100}{104}, \frac{100}{50}] \tag{5.2.4}$$

We used the best hyper-parameters that was obtained for the softness assessment task. Accordingly, the drop-rate was set to 0, weight regularization coefficient was 10^{-4}, and the color-jitter rate was 0.2. A sample result corresponding to one of the folds of the training run is shown in Figure 5.6. Similar to softness assessment, we could not observe a consistently decreasing loss (or increasing recall). For the given example, we selected the 5^{th} fold as it produced the highest recall rate.

The overall results of the 10-fold cross validation are summarized in Tables 5.11, 5.12, and 5.13 respectively for individual patch, mean, and majority voting results. The corresponding confusion matrices are shown in Figure 5.7. These matrices are also predominantly diagonal indicating that even if prediction errors are made they are

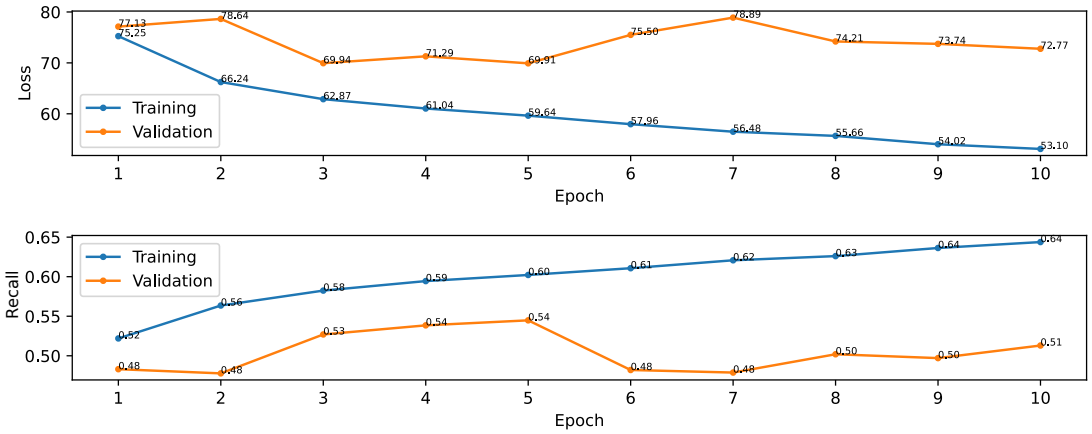


Figure 5.6: Change of mean squared error and average recall rates during one fold of the training for the pilling assessment task.

Metric	Per-class value (1, 2, 3, 4, 5)
Precision	0.35, 0.55, 0.37, 0.56, 0.52
Recall	0.20, 0.63, 0.31, 0.64, 0.43
F1 Score	0.25, 0.59, 0.34, 0.60, 0.47

Table 5.11: Precision, recall, and F1 score results for deep learning based classification on the pilling assessment task. Results of classifying each patch separately are shown in this table.

made mostly to neighboring classes. Furthermore, similar to the softness assessment, classes for which we had a higher number of samples received better classification scores. The best classification scores were obtained for class 4 for which we had 104 samples. In contrast, the worst classification scores were obtained for class 1, for which we had only 14 samples. As a result, almost all class 1 samples were predicted to be class 2 as can be observed from the deep blue color in the first row of the confusion matrices.

Finally, we share the overall accuracy results in Table 5.14 and compare them with those of the softness assessment task (Table 5.10). It can be observed from both tables that the accuracy scores for pilling assessment are higher than the corresponding scores for softness assessment. The best result was obtained for the mean approach (0.60) where the mean class values of individual patches are averaged to obtain the

Metric	Per-class value (1, 2, 3, 4, 5)
Precision	$0.20, 0.60, 0.52, 0.64, 0.62$
Recall	$0.07, 0.65, 0.53, 0.77, 0.40$
F1 Score	$0.10, 0.62, 0.52, 0.70, 0.49$

Table 5.12: Precision, recall, and F1 score results for deep learning based classification on the pilling assessment task. Results based on averaging patch classes for each image are shown.

Metric	Per-class value (1, 2, 3, 4, 5)
Precision	$0.20, 0.56, 0.40, 0.61, 0.62$
Recall	$0.07, 0.76, 0.20, 0.80, 0.42$
F1 Score	$0.10, 0.64, 0.27, 0.69, 0.50$

Table 5.13: Precision, recall, and F1 score results for deep learning based classification on the pilling assessment task. Results based on majority voting of patch classes for each image are shown.

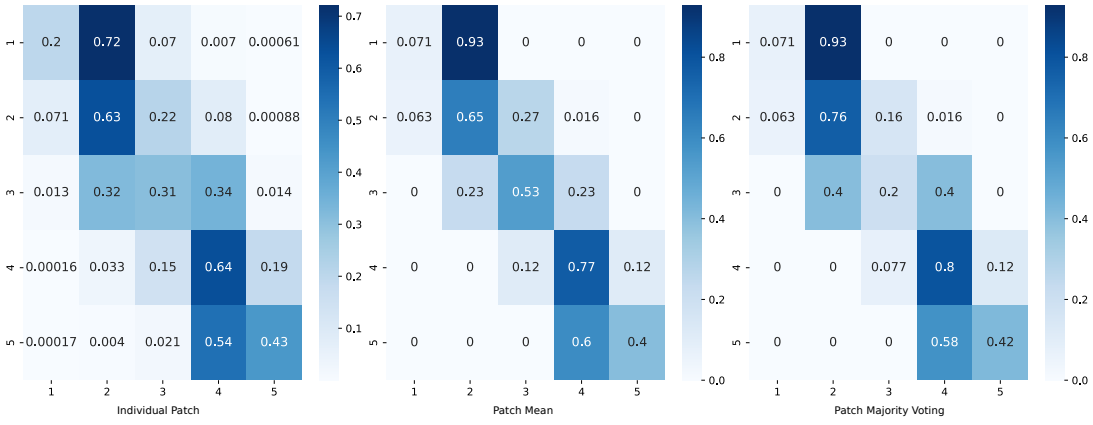


Figure 5.7: Confusion matrices for individual patch, patch mean, and patch majority voting results for the pilling assessment task.

Approach	Accuracy
Individual patch	0.51
Mean patch per image	0.60
Majority vote patch per image	0.57

Table 5.14: Overall accuracy of different prediction approaches for pilling assessment using deep learning.

class score for a high resolution sample.

CHAPTER 6

GRAPHICAL USER INTERFACE

6.1 General Overview and Features

To make the feature analysis procedure easier for the users, we developed a graphical user interface (GUI). The PyQt framework was used to implement the user interface chosen because of the PyTorch environment. The main UI has four different tabs, each for different major functionalities. The GUI has additional features such as localization, personalization, training parameter controls, and help menus. In Figure 6.1 and Figure 6.2 two different theme options are demonstrated. There are seven different themes to make usage easy depending on user preference. In Figure 6.3 the example parameter setting is demonstrated. Furthermore, users are informed about the ongoing process through the progress bar and details panel at the bottom of the GUI. Moreover, the training and testing result is visualized as a graph in relevant tabs.

6.2 Data Preparation

The first screen of the GUI is the dataset generation tab. In this tab, the program enables users to generate training data quickly. To create a training dataset, the user must input the images' directory path, the output path for the generated dataset, the percentages of the test, and the validation data. The input data directory should contain five different directories with the name of the class values (1,2,3,4,5). The output dataset generates the output path under an automatically created train, test, and validation data folder. Progress of the data creation process is visualized through the progress bar in the center of the window and the more detailed outputs in the particu-

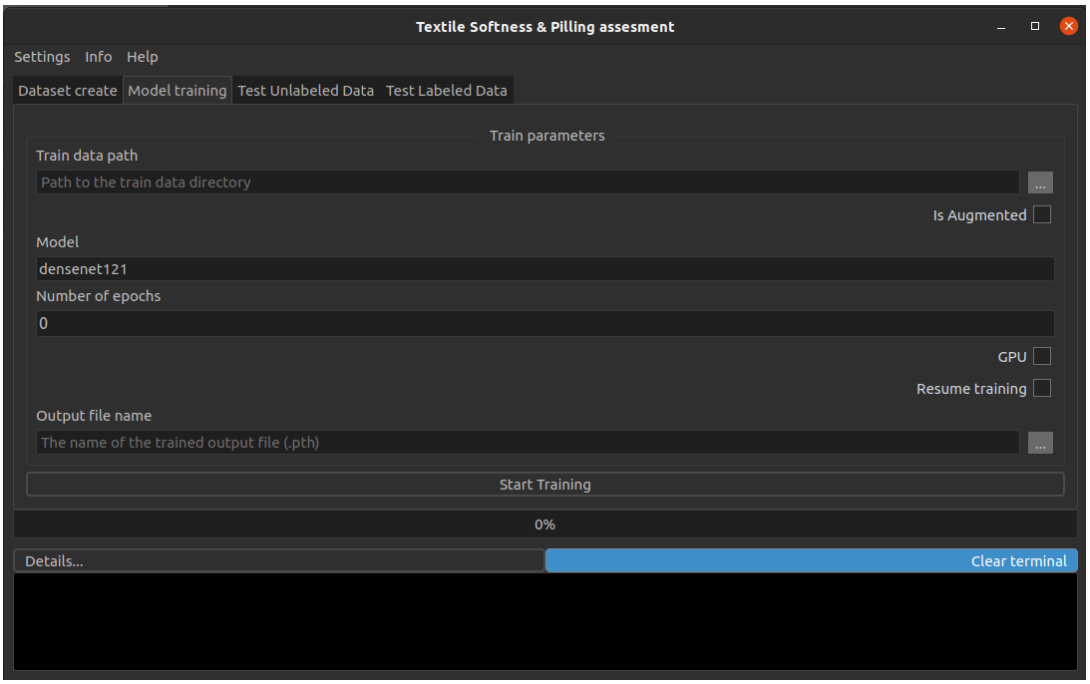


Figure 6.1: Example theme demonstration.

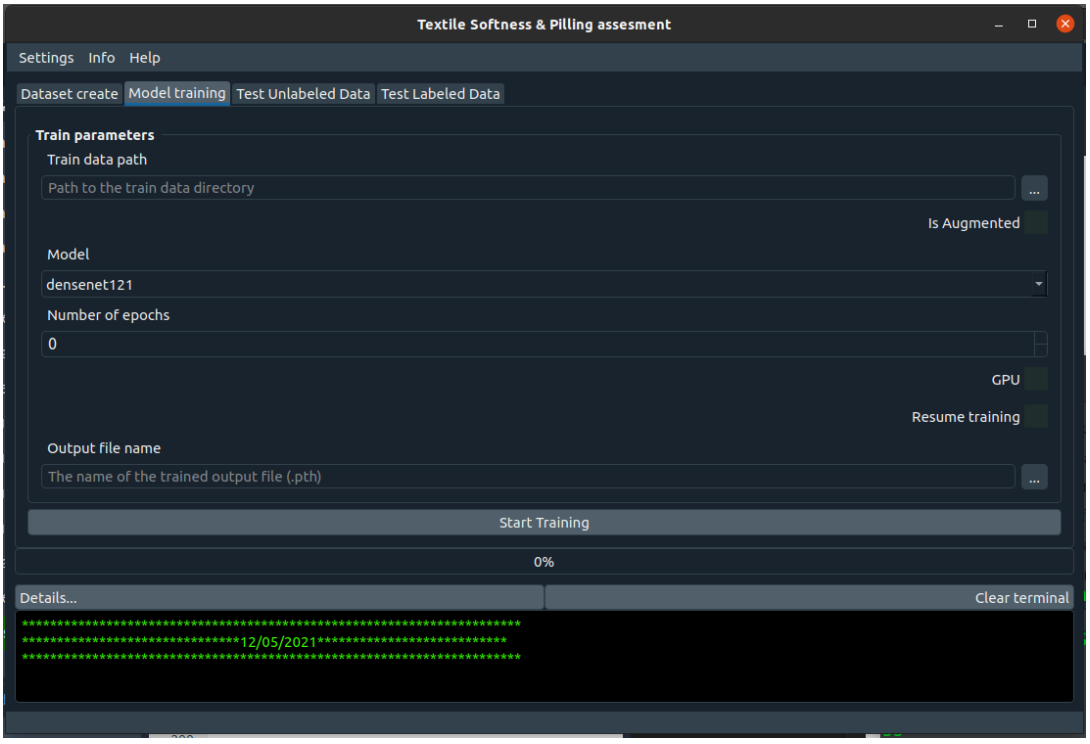


Figure 6.2: Another theme demonstration.

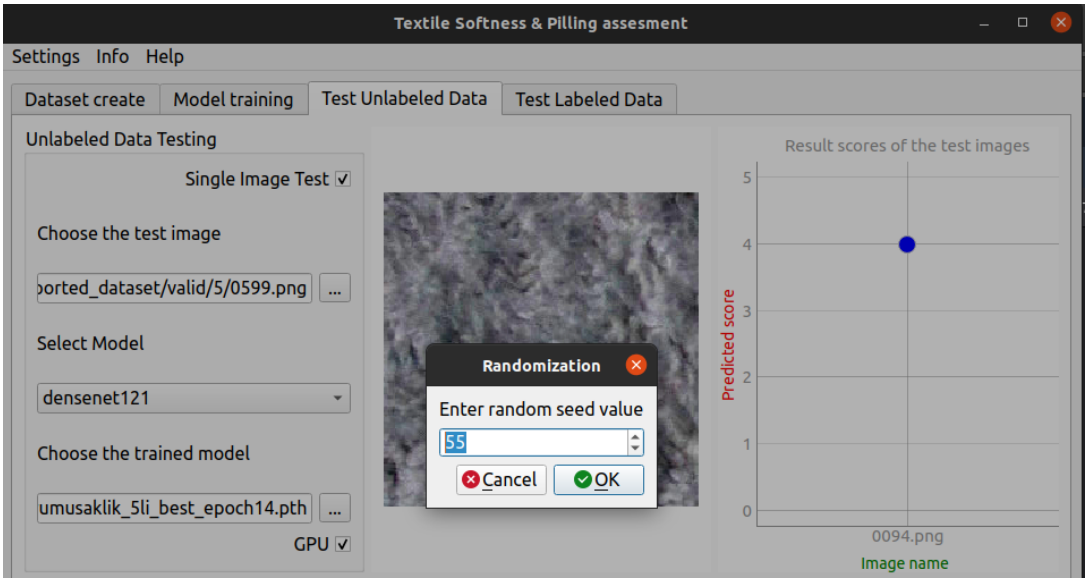


Figure 6.3: Setting the random seed value for graphical user interface.

lar area of the bottom of the window.

6.3 Training Tab

To finetune PyTorch's pre-trained models with the dataset created in the dataset generate tab, the user can quickly fill some parameters and start the training process. If the user has already generated training data, the path of the created data is automatically populated into the data path input field on the training tab. It is also possible to choose pre-generated data from the file system. The user also needs to select the network name from the drop-down menu where most of the pre-trained networks of the PyTorch are included.

Furthermore, the user must input the epoch count and the output file name for the trained model. If the output file name is not set, the program creates the output file in the current directory with the default file name. To finish the training procedure, the user needs to start the training after filling all the fields. The details of the training process (training and validation losses, training progress) is visualized during the training process. The final result is also visualized on the right side of the tab. The procedure has been demonstrated in Figures 6.4, 6.5, and 6.6.

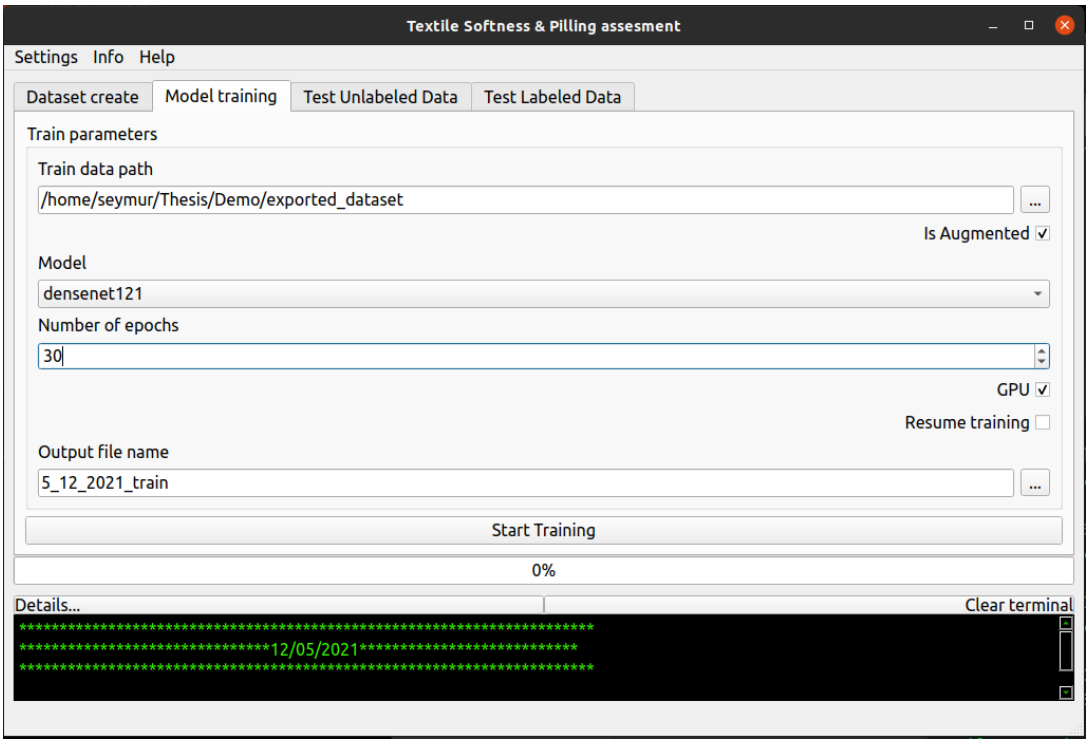


Figure 6.4: Training tab.

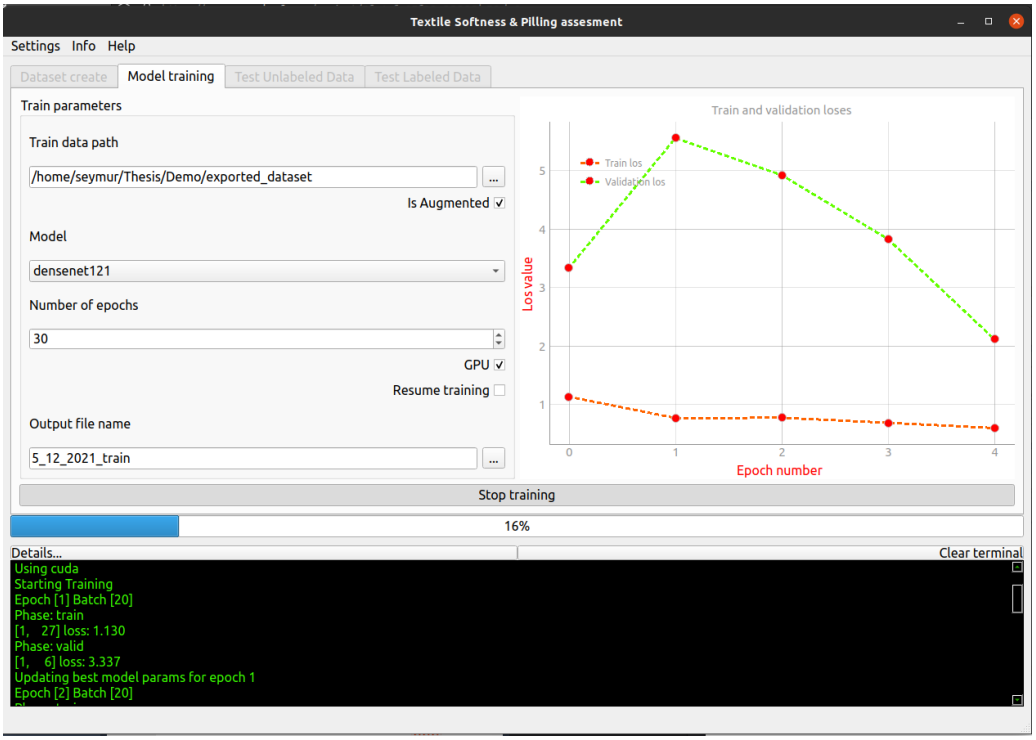


Figure 6.5: Training tab.

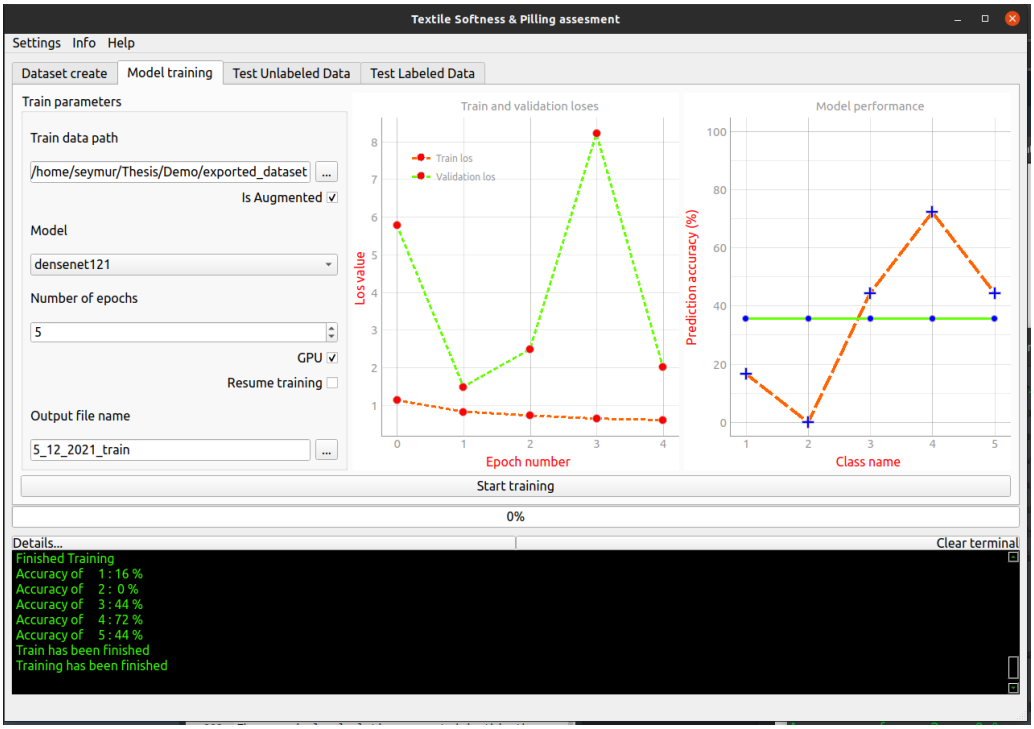


Figure 6.6: Training tab.

6.4 Test Unlabeled Data Tab

One of the main parts of the developed GUI is a tab to test the softness and pilling measures of the unlabeled data using a pre-trained model file. The user must fill the predefined input fields to start testing, namely the input images' directory field, trained model path field, the output file path, and the drop-down menu to select the trained model name. There is a checkbox that is optional to decide whether to use GPU utilization during the procedure or not.

There are two modes to test the unlabeled data. The first mode is to test a single image, which is demonstrated in Figure 6.8. The second mode is to test multiple unlabeled data in which it is also possible to export the result into a CSV file. In Figure 6.9 six textile examples, which have a softness value of 4 were tested, and the result are demonstrated as a chart on GUI.

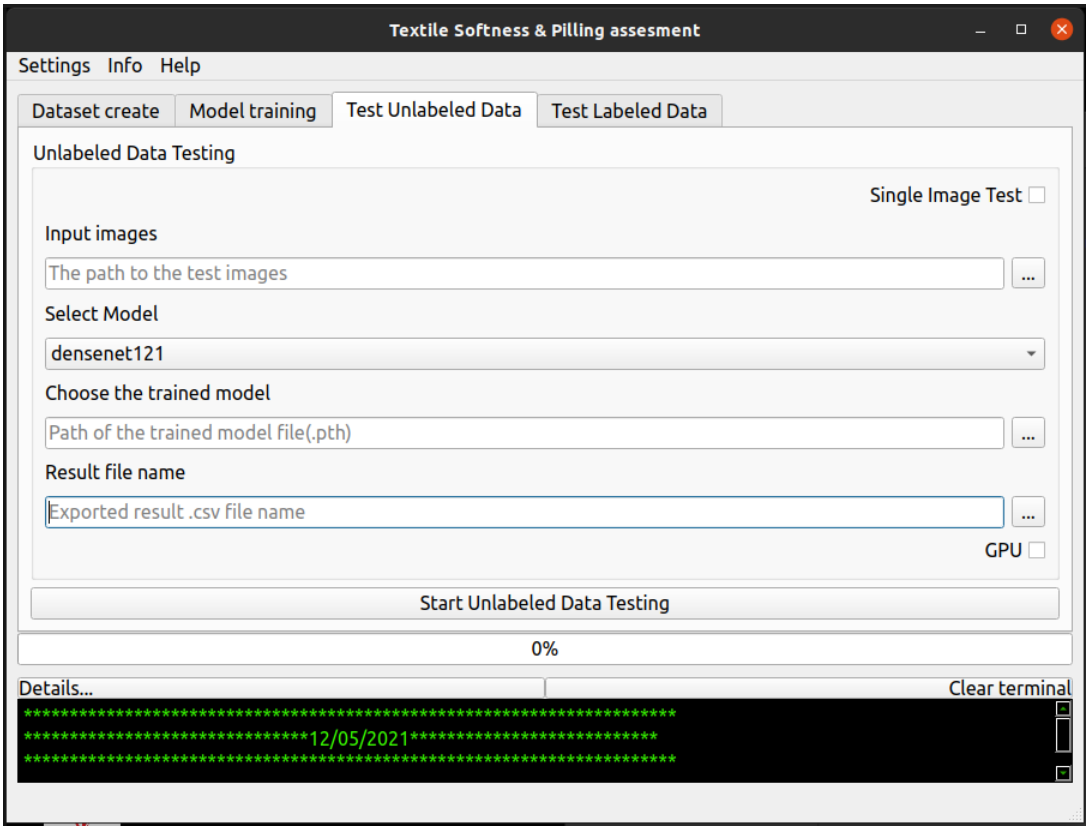


Figure 6.7: The initial view of the tab to test unlabeled data.

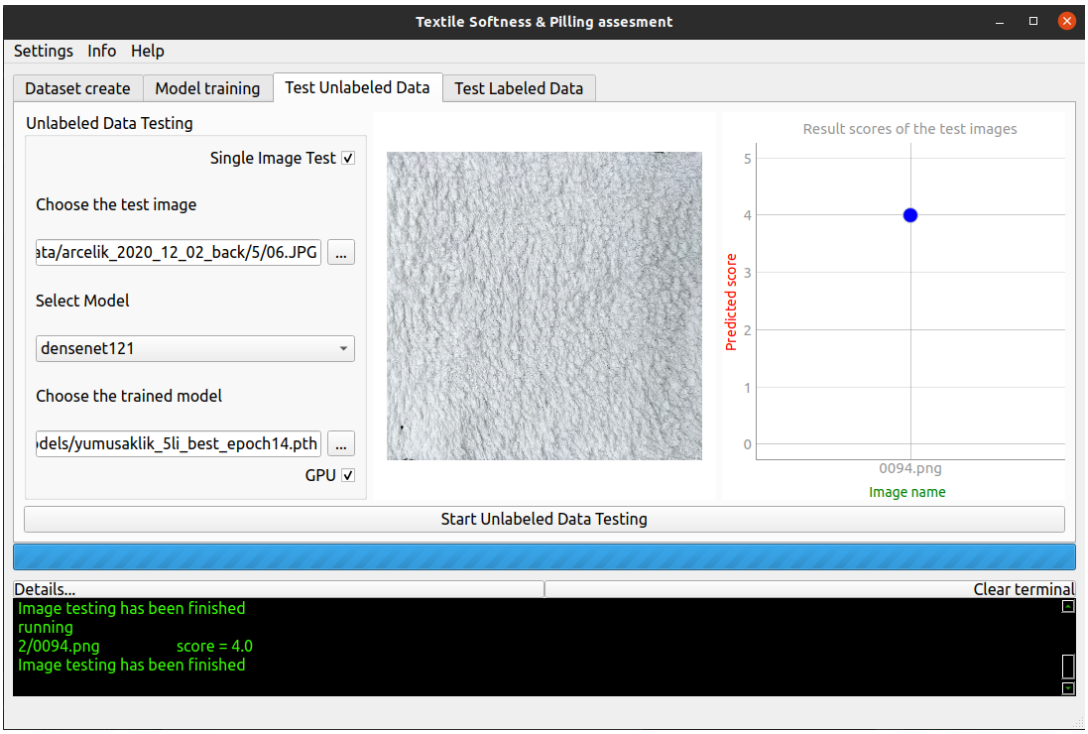


Figure 6.8: The result of analyzing single unlabeled data.

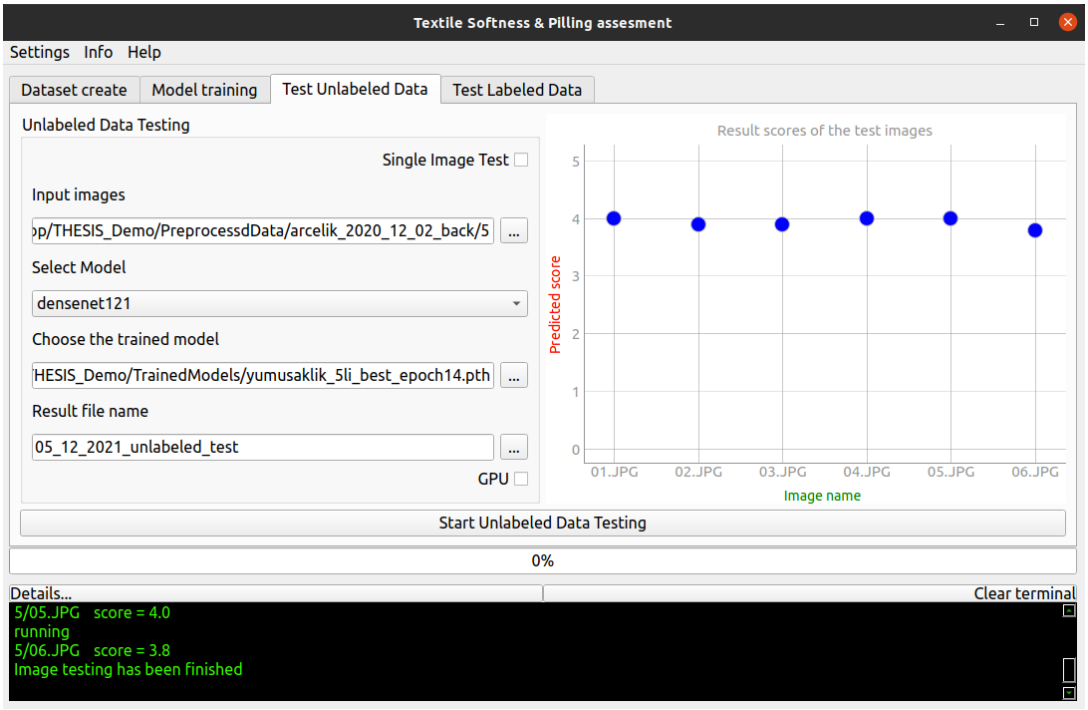


Figure 6.9: The result of analyzing multiple unlabeled data.

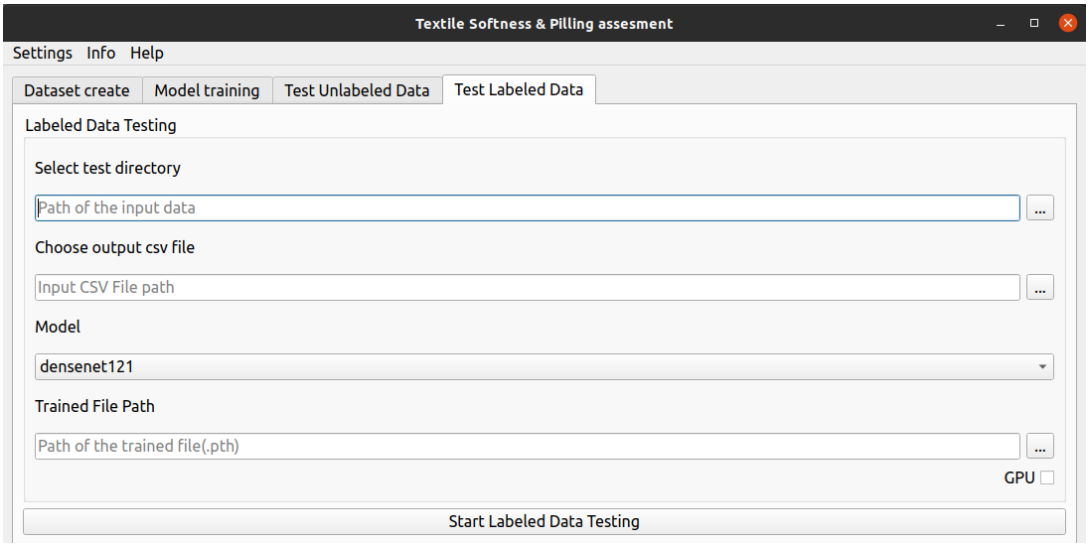


Figure 6.10: The initial view of the tab to test labeled data.

6.5 Test Labeled Data Tab

The last tab is to test the softness and pilling measure of the labeled data. To start testing, the user must fill predefined input fields, including the input images directory field, trained model path field, and the drop-down menu to select the pre-trained model name. The user is also needed to fill the input CSV file, which includes values of the scores of the input images to compare with the test result. Furthermore, there is a checkbox to enable or disable GPU utilization for the procedure. In Figure 6.10, the general overview of the labeled data test tab is visualized.

In summary, the classification algorithms developed in this study are implemented under an easy-to-use GUI to enable their use by end-users who are not familiar with coding and deep learning. The GUI is given to the Arcelik team to be used in their facility.

CHAPTER 7

CONCLUSIONS

In this study, we proposed a new method to automatically classify fabric samples into different softness and pilling classes. In our study, we first evaluated the effectiveness of hand-crafted features for these tasks. To this end, we tested 13 HCFs that that involve basic features, Tamura features, and GLCM features. We first tested the correlation of these features with the expert scores. We used four different classifiers namely linear regression, multinomial logistic regression, ordinal logistic regression, and support vector machines. The classification test was done only for the softness assessment. This is because the tested features did not show significant correlation with the expert scores for the pilling task.

Our overall conclusion for this part of the study is that while HCFs show some promise for the softness assessment task, their performance for pilling assessment is dubious. Perhaps other features that were not tested in this study could be more suitable for pilling assessment. As for the classifiers, we found the multinomial regression to have the best performance although the ordinal regression and SVM scores were not significantly different.

In the next phase of our study, we experimented with deep learning methods as they are commonly used for classification tasks. However, it was not clear in the beginning of this study how well they will perform for softness and pilling assessment. This is because the images for these assessments are very different from standard object classification, in which the objects much more different from each other. Note that experts who judge make these judgements do not based their decisions only on the visuals. They touch and feel the fabrics, which is something that cannot be done on the computer environment. Therefore, we did not have a clear picture of how well

deep learning models would perform just based on the images of the samples.

Our results yielded several interesting findings. First we found that deep learning based methods outperform traditional classifiers that are based on hand-crafted features. Secondly, we found that DL methods perform slightly better for pilling assessment compared to softness assessment. Our best DL based classifier for pilling had an accuracy of 0.60, which is significantly above the chance level of 0.20 for a five class classification task. We also observed that the number of samples plays a key role for the success of a DL-based classifier. We consistently observed that the classes with higher number of samples received better classification accuracy compared to those that had fewer number of samples. Indeed, DL-based methods are known to be data hungry and we witnessed this problem in the context of this thesis as well.

We can perhaps argue that with the current amount of training data, aiming to perform a five class classification is a too difficult task even for DL-based methods. However, as evidenced by the visualized confusion matrices, the errors they make are not huge. As such, with the current dataset (154 images for softness and 291 for pilling), a three class classification problem can be more realistic. However, if a higher number of training images are used, a five class classification appears to be achievable.

The domain knowledge as well as the dataset for this study was kindly provided by Arcelik. In addition to several scientific findings, as a practical contribution of this study, we developed a GUI that can be used in Arcelik laboratories for automated softness assessment. Several training sessions have been conducted to teach the use of this GUI to Arcelik engineers. We hope that the availability of such a program can help motivate further work for automated fabric quality assessment.

7.1 Possible Use Cases

This study, in its current form, could be used in several different ways. The most prominent one is integrating the algorithm in the fabric manufacture factory environment and measuring the pilling characteristics of the nearly identical fabric pieces to decide on the fabric pilling durability before mass production. This would be especially appropriate for some of the fabric types, such as cotton knitted and woven

fabrics.

As another use case, manufactactures of washing machines want to know after how many washing cycles the qualities of a fabric start to change. This is also related to detergent and softener use. So the algorithm developed in this study can be used by both manufacturers like Arcelik or producers of detergents and softeners (i.e. chemical companies) to assess the change a fabric undergoes during its life cycle. It can help them make characterize their equipment and make more appropriate changes to their customers. While some studies are already being performed, as they require manual labor they are costly in terms of both time and man-power. Our study can significantly simplify this process.

As an ultimate use case for this study, a camera placed in a water-tight compartment within the washing machine can analyze the overall softness and pilling degrees of the clothes that are placed in the machine. In a dry cycle that can take a few seconds, several images of the clothes can be be captured and then analyzed. Based on the results of this analysis, the amount of softener and detergent can automatically be adjusted. Such as solution would have an essential environmental and marketing value. While we think that this is not a simple problem, the ground work laid out in this thesis can be considered as the first step in this direction.

7.2 Limitations

The lack of the features to measure the softness on different fabrics with different materials and textures can be considered as the current study's chief limitation: we only used images of towels and socks with a certain color. Therefore, it is tough to generalize the results to thousands of other fabric types with different colors, materials, and textures. The dataset is fundamental limiting factor to generalize our methodology to the wide range of fabric sets. We need different fabric groups to classify each set separately. The distance that the input images have been captured is another limitation of the work, which should be approximately the same for every image. These controlled conditions may be difficult to achieve in out-of-laboratory environments.

7.3 Future Work

The first idea that comes to mind as future work is to detect the softness and pilling of the fabric in a natural environment. As a result of our research and experiments, we have seen that it is possible to detect the softness and pilling of the material by applying our methods. The natural environment, such as the washing cabin of the washing machine, should be considered the final destination or an integration environment of the system.

Another future work should be improving the algorithms to work with different textured and different materialized fabrics. We need a massive amount of data in the dataset for DL implementation to train our models to implement different kinds of fabrics with different materials and textures.

Combining hand-crafted and deeply learned features to generate a hybrid method of these two approaches for measuring the fabric's softness and pilling value and comparing the result with the results of the separated methods may be one of the other possible future directions.

REFERENCES

[1] H. Tamura, S. Mori and T. Yamawaki, *"Textural Features Corresponding to Visual Perception."* in IEEE Transactions on Systems, Man, and Cybernetics, vol. 8, no. 6, pp. 460-473, June 1978, doi: 10.1109/TSMC.1978.4309999.

[2] L. Hsin-Chin, C. Chih-Yi, and Y. Shi-Nine, *"Textural Analysis and Description in Linguistic Terms"*. ACCV2002: The 5th Asian Conference on Computer Vision, 23-25 January 2002, Melbourne, Australia

[3] R huertas, Shunji Mori, and Takashi Yamawaki. *Texture Characterization based on Grey-Level Co-occurance Matrix*. (German) [*On the electrodynamics of moving bodies*]. Annalen der Physik, 322(10):891–921, 1905.

[4] Haralick, Robert & Shanmugam, K. & Dinstein, Ih. (1973). *"Textural Features for Image Classification"*. IEEE Trans Syst Man Cybern. SMC-3. 610-621.

[5] L. -. Soh and C. Tsatsoulis, *"Texture analysis of SAR sea ice imagery using gray level co-occurrence matrices,"* in IEEE Transactions on Geoscience and Remote Sensing, vol. 37, no. 2, pp. 780-795, March 1999, doi: 10.1109/36.752194.

[6] Echoview Help file 11.1.01 for Echoview 11.1.23, (16 October, 2020). *"GLCM Texture Feature"*. Retrieved December 14, 2020, from : https://support.echoview.com/WebHelp/Windows_and_Dialog_Boxes/Dialog_Boxes/Variable_properties_dialog_box/Operator_pages/GLCM_Texture_Features.htm#Correlation

[7] American Journal of Neuroradiology, (n.d). *"Formulas and description of GLCM texture parameters"*. Retrieved December 14, 2020, from : http://www.ajnr.org/content/suppl/2015/09/10/ajnr.A4455.DC1/15-00131.pdf

[8] R. Vyas, T. Kanumuri, G. Sheoran and P. Dubey, *"Co-occurrence features and neural network classification approach for iris recognition,"* 2017 Fourth Inter-

national Conference on Image Information Processing (ICIIP), Shimla, 2017, pp. 1-6, doi: 10.1109/ICIIP.2017.8313730.

[9] Bahl, K. Singh Kaith, J. *"Evaluation of Yarn Quality in Fabric using Image Processing Techniques."*. International Journal of Science and Research (IJSR) ISSN (Online): 2319-7064

[10] S. Roy, A. Sengupta and S. Sengupta, *"Yarn hairiness evaluation using image processing,"* Proceedings of The 2014 International Conference on Control, Instrumentation, Energy and Communication (CIEC), Calcutta, 2014, pp. 588-592, doi: 10.1109/CIEC.2014.6959157.

[11] Stojanovic, R., Mitropulos, P., Koulamas, C., Karayiannis, Y., Koubias, S., *"Real-time vision-based system for textile fabric inspection."*. Papadopoulos, G. (2001). Real-Time Imaging(Vol. 7, Issue 6). https://doi.org/10.1006/rtim.2001.0231

[12] Grüner. A, *"Emtec TSA – Textile Softness Analyzer: A new and objective way to measure smoothness, softness and stiffness of textiles."*. Global Marketing and Business Development Manager emtec Electronic GmbH – Leipzig, Germany 09.04.2018

[13] Schlomski. I, *"The softness of fabrics."*. https://textile-network.com/en/Technical-Textiles/Technologien/The-softness-of-fabrics Emtec Electronic at the Techtextil Hall 3.1, stand C 77 25/04/2019

[14] Ruel, E,T., *"Method and apparatus for softness testing."*. United States Patent Aug 15 1972

[15] A. Das, R. Alagirusamy *"Science in clothing comfort."*. IIT Delhi, 2010

[16] P. Zhang, L. Xin, W. Lijing, W. Xungai, *"An experimental study on fabric softness evaluation"*. College of Textiles, Donghua University, Shangai, People's Republic of China and School of Engineering and Technology, Deakin University, Deakin, Australia October 2005

[17] L. Yuzheng, G. Weidong, W. Ya and L. Jihong, *"The objective evaluation method for the woven fabric Softness grading,"* 2009 IEEE 10th International

Conference on Computer-Aided Industrial Design & Conceptual Design, Wenzhou, 2009, pp. 2232-2237, doi: 10.1109/CAIDCD.2009.5375203.

[18] X. Meng, W. Zhang and S. Cong, *"The Objective Evaluation Model on Wearing Touch and Pressure Sensation Based on GRNN,"* 2010 Third International Conference on Information and Computing, Wuxi, 2010, pp. 289-292, doi: 10.1109/ICIC.2010.168.

[19] M. Eldessouki, *"7 - Evaluation of fabric pilling as an end-use quality and a performance measure for the fabrics"*, Editor(s): W.K. Wong, In The Textile Institute Book Series, Applications of Computer Vision in Fashion and Textiles, Woodhead Publishing, 2018, Pages 147-187,

[20] C. Zhi, Z. Gao, G. Wang, M. Chen, W. Fan and L. Yu, *"Fabric Pilling Hairiness Extraction From Depth Images Based on the Predicted Fabric Surface Plane,"* in IEEE Access, vol. 8, pp. 5160-5171, 2020, doi: 10.1109/ACCESS.2019.2962917.

[21] K. Soo Chang and T. Jin Kang, *"Automatic Evaluation of Fabric Pilling Using a 3-D Non-contact Scanning System,"* 2005 IEEE Instrumentationand Measurement Technology Conference Proceedings, Ottawa, Ont., 2005, pp. 628-632, doi: 10.1109/IMTC.2005.1604193.

[22] X. Zengbo and Y. Hongsui, *"Fabric Pilling Object Detection Based on Scale - Space Extremum,"* 2015 2nd International Conference on Information Science and Control Engineering, Shanghai, 2015, pp. 229-233, doi: 10.1109/ICISCE.2015.58.

[23] C. Ren, S. Zheng, H. Liu and Y. Han, *"Evaluation of fabric pilling based on Hough transform and Gabor filter,"* 2017 First International Conference on Electronics Instrumentation & Information Systems (EIIS), Harbin, 2017, pp. 1-4, doi: 10.1109/EIIS.2017.8298613.

[24] Z. Xiao and H. Yang, *"Fabric Pilling Segmentation Based on Edgeflow Algorithm,"* 2007 International Conference on Machine Learning and Cybernetics, Hong Kong, 2007, pp. 1744-1748, doi: 10.1109/ICMLC.2007.4370429.

[25] L. Xiaojun, *"Segmentation for Fabric Pilling Images Based on Edge Flow,"* 2009 Second International Conference on Information and Computing Science, Manchester, 2009, pp. 369-372, doi: 10.1109/ICIC.2009.204.

[26] Z. Xiao, J. Wu, L. Geng, J. Wang, N. Xu and Z. Lin, *"Objective Assessment of Pilling of Knitted Fabrics Based on Improved BP Neural Network and Genetic Algorithm,"* 2009 Fifth International Conference on Natural Computation, Tianjin, 2009, pp. 222-225, doi: 10.1109/ICNC.2009.179.

[27] Huang, Mei-Ling & Fu, Chien-Chang. (2018). *"Applying Image Processing to the Textile Grading of Fleece Based on Pilling Assessment."* Fibers. 6. 73. 10.3390/fib6040073.

[28] Namlıgöz Dalbaşı, Eylen & Kayseri, G.. (2015). *"A research about the effect of the anti-pilling treatments on different structured cotton knitted fabrics."* Tekstil ve Konfeksiyon. 25. 54-60.

[29] Gültekin, Elif & Çelik, Halil & Dülger, Lale & Sünbül, H & Kani, H. (2019). *"IMAGE PROCESSING APPLICATIONS ON YARN CHARACTERISTICS AND FAULT INSPECTION."*

[30] Ribeiro, Rui & Pilastri, André & Moura, Carla & Rodrigues, Filipe & Rocha, Rita & Morgado, José & Cortez, Paulo. (2020). *"Predicting Physical Properties of Woven Fabrics via Automated Machine Learning and Textile Design and Finishing Features."* 10.1007/978-3-030-49186-4-21.

[31] Mokwińska, Z.. (2006). *"Determination of fabric propensity to surface fuzzing and to pilling by interlaboratory comparison according to PN-EN ISO 12945:2002."* 60. 33-37+2.

[32] Sparavigna, Amelia Carolina. (2017). *"Image Segmentation Applied to the Analysis of Fabric Textures."*

[33] Wang, Xin & Georganas, Nicolas & Petriu, Emil. (2011). *"Fabric Texture Analysis Using Computer Vision Techniques."* Instrumentation and Measurement, IEEE Transactions on. 60. 44 - 56. 10.1109/TIM.2010.2069850.

[34] Pan, Ruru & Gao, Weidong & Liu, Jihong & Wang, Hongbo. (2011). *"Automatic recognition of woven fabric pattern based on image processing*

and BP neural network." The Journal of the Textile Institute. 102. 19-30. 10.1080/00405000903430255.

[35] Iqbal Hussain MA, Khan B, Wang Z, Ding S. (2020). *"Woven Fabric Pattern Recognition and Classification Based on Deep Convolutional Neural Networks."* Electronics. 2020; 9(6):1048.

[36] Karsten L, Janosch A, Dirk W. S, *"Novel objective test method for the abrasion and pilling behaviour of low basis weight spunbond polypropylene nonwovens"* Polymer Testing, Volume 69, 2018, pp 175-181, ISSN 0142-9418, doi: 10.1016/j.polymertesting.2018.05.019.

[37] El-Dessouki, D.H. *"A Study on Abrasion Characteristics and Pilling Performance of Socks"* International Design journal, Volume 4 Issue 2, 2014

[38] N., Özdil, G., Özçelik, Süpüren Mengüç, Gamze. *"Abrasion Resistance of Materials"* Chapter 7: Analysis of Abrasion Characteristics in Textiles.(2012).Janeza Trdine 9, 51000 Rijeka, Croatia

[39] NC STATE UNIVERSITY *"Kawabata Evaluation System"* Textile Protection And Comfort Center https://textiles.ncsu.edu/tpacc/comfort-performance/kawabata-evaluation-system/

[40] P. Sengotuvelan, A. Wahi, and A. Shanmugam, *"Automatic fault analysis of textile fabric using imaging systems."*. Research Journal of Applied Science, vol. 3, pp. 26-31, 2008

[41] P. Sengotuvelan, A. Wahi, and A. Shanmugam, *"Classification of Fabric Defects"*. https://www.onlineclothingstudy.com/2019/02/classification-of-fabric-defects.html Research Journal of Applied Science, vol. 3, pp. 26-31, 2008

[42] Cotton Incorporated, (n.d). *"Standard Fabric Defect Glossary"*. Retrieved December 26, 2020, from : https://www.cottoninc.com/quality-products/textile-resources/fabric-defect-glossary/

[43] K. Hanbay, M. F. Talu, Ö. F. Özgüven and D. Öztürk, *"Fabric defect detection methods for circular knitting machines,"* 2015 23nd Signal Processing and Communications Applications Conference (SIU), Malatya, 2015, pp. 735-738, doi: 10.1109/SIU.2015.7129932.

[44] H. Zhang, L. Zhang, P. Li and D. Gu, *"Yarn-dyed Fabric Defect Detection with YOLOV2 Based on Deep Convolution Neural Networks,"* 2018 IEEE 7th Data Driven Control and Learning Systems Conference (DDCLS), Enshi, 2018, pp. 170-174, doi: 10.1109/DDCLS.2018.8516094.

[45] W. Jianxia, Z. Guang, M. Hongxia and Q. Min, *"Feature Extraction of the Fabric Defects Image,"* 2007 8th International Conference on Electronic Measurement and Instruments, Xi'an, 2007, pp. 2-714-2-717, doi: 10.1109/ICEMI.2007.4350781.

[46] X. Zhang, W. Xu, R. Pan, J. Liu and W. Gao, *"Fabric defect detection based on projected transform for feature extraction,"* 2011 10th International Symposium on Distributed Computing and Applications to Business, Engineering and Science, Wuxi, 2011, pp. 192-196, doi: 10.1109/DCABES.2011.42.

[47] K. Sakhare, A. Kulkarni, M. Kumbhakarn and N. Kare, *"Spectral and spatial domain approach for fabric defect detection and classification,"* 2015 International Conference on Industrial Instrumentation and Control (ICIC), Pune, 2015, pp. 640-644, doi: 10.1109/IIC.2015.7150820.

[48] O. Kaynar, Y. E. Işik, Y. Görmez and F. Demirkoparan, *"Fabric defect detection with LBP-GLMC,"* 2017 International Artificial Intelligence and Data Processing Symposium (IDAP), Malatya, 2017, pp. 1-5, doi: 10.1109/IDAP.2017.8090188.

[49] Postle, R., Carnaby, G. A., & De Jong, S. *"The Mechanics of Wool Structures",* Chapt, 14, pp. 369-385, John Willey & Sons. (1989)

[50] Morton W. E., Hearl J. W. S.,; *"Physical Properties of Textile Fibers",*. 1973.

[51] Mirjalili S. A., Ekhtiyari ,; *"Wrinkle Assesment of Fabric Using Image Processing",*. Textile Engineering Department, Yazd University Iran 2010.

[52] C. Turner, H. Sari-Sarraf, Aijun Zhu, E. Hequet and Sunho Lee, *"Automatic assessment of fabric smoothness,"* The 2002 45th Midwest Symposium on Circuits and Systems, 2002. MWSCAS-2002., Tulsa, OK, USA, 2002, pp. II-II, doi: 10.1109/MWSCAS.2002.1186877.

[53] A. D. Rosli, H. Hashim, N. E. Abdullah, N. Ismail and Z. Halil, *"Index on fabric wrinkle using image processing technique,"* 2012 IEEE Control and System

Graduate Research Colloquium, Shah Alam, Selangor, 2012, pp. 186-190, doi: 10.1109/ICSGRC.2012.6287159.

[54] Statistical Solutions (n.d.). *"Correlation (Pearson, Kendall, Spearman)"* Retrieved December 20, 2020, from https://www.statisticssolutions.com/correlation-pearson-kendall-spearman/

[55] Barcelona Field Studies Center (n.d.). *"Spearman's Rank Correlation Coefficient"* Retrieved December 20, 2020, from https://geographyfieldwork.com/SpearmansRank.htm

[56] Spearman, C. (1987). *"The proof and measurement of association between two things."* The American journal of psychology, 100(3/4), 441-471.

[57] Sedgwick, Philip. (2014). *"Spearman's rank correlation coefficient."* BMJ (online). 349. g7327. 10.1136/bmj.g7327.

[58] Freund, Rudolf J and Wilson, William J and Sa, Ping *"Regression analysis"* Elsevier, (2006)

[59] Paszke, Adam and Gross, Sam and Chintala, Soumith and Chanan, Gregory and Yang, Edward and DeVito, Zachary and Lin, Zeming and Desmaison, Alban and Antiga, Luca and Lerer, Adam *"Automatic differentiation in PyTorch"* NIPS-W, 2017

[60] PyQt *"PyQt Reference Guide"* , 2012

[61] M. Summerfield, (2007) *"Rapid GUI programming with Python and Qt : the definitive guide to PyQt programming."* , Upper Saddle River, NJ :Prentice Hall,

[62] Van Rossum, Guido and Drake Jr, Fred L *"Python tutorial"*, Centrum voor Wiskunde en Informatica Amsterdam, The Netherlands, 1995

[63] M. Sandler, A. Howard, M. Zhu, A. Zhmoginov and L. Chen, *"MobileNetV2: Inverted Residuals and Linear Bottlenecks."* 2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2018, pp. 4510-4520, doi: 10.1109/CVPR.2018.00474.

[64] Huang, G., Liu, Z., van der Maaten, L. & Weinberger, K. Q. (2017).*"Densely Connected Convolutional Networks."*, CVPR (p./pp. 2261-2269), : IEEE Computer Society. ISBN: 978-1-5386-0457-1

[65] Krizhevsky, Alex & Sutskever, Ilya & Hinton, Geoffrey E. (2012).*"ImageNet Classification with Deep Convolutional Neural Networks."*, Curran Associates, Inc. (V25).

[66] C. Szegedy et al., (2015).*"Going deeper with convolutions."*, 2015 IEEE Conference on Computer Vision and Pattern Recognition. (CVPR), 2015, pp. 1-9, doi: 10.1109/CVPR.2015.7298594.

[67] Simonyan, Karen & Zisserman, Andrew. (2014).*"Very Deep Convolutional Networks for Large-Scale Image Recognition."* arXiv 1409.1556.

[68] Iandola, Forrest & Moskewicz, Matthew & Ashraf, Khalid & Han, Song & Dally, William & Keutzer, Kurt. (2016, February).*"SqueezeNet: AlexNet-level accuracy with 50x fewer parameters and textless1MB model size."*

[69] Zhang, Xiangyu & Zhou, Xinyu & Lin, Mengxiao & Sun, Jian. (2018, June).*"ShuffleNet: An Extremely Efficient Convolutional Neural Network for Mobile Devices."* doi: 10.1109/CVPR.2018.00716 6848-6856

[70] Howard, Andrew & Zhu, Menglong & Chen, Bo & Kalenichenko, Dmitry & Wang, Weijun & Weyand, Tobias & Andreetto, Marco & Adam, Hartwig. (2017, April).*"MobileNets: Efficient Convolutional Neural Networks for Mobile Vision Applications."*

[71] C. Szegedy et al., (2015).*"Going deeper with convolutions."* 2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR) 1-9 doi: 10.1109/CVPR.2015.7298594

[72] Napoletano, Paolo & Piccoli, Flavio & Schettini, Raimondo. (2018). *"Anomaly Detection in Nanofibrous Materials."* Sensors (Basel, Switzerland). 18. 10.3390/s18010209.

[73] Deng, J., Dong, W., Socher, R., Li, L. J., Li, K., & Fei-Fei, L. (2009, June). *"Imagenet: A large-scale hierarchical image database."* In 2009 IEEE conference on computer vision and pattern recognition (pp. 248-255). IEEE.

[74] Cheng, J., Wang, Z., & Pollastri, G. (2008, June). *"A neural network approach to ordinal regression."* In 2008 IEEE international joint conference on neural networks (IEEE world congress on computational intelligence) (pp. 1279-1284). IEEE.

[75] Pedregosa, F., Varoquaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O., ... & Duchesnay, E. (2011). *"Scikit-learn: Machine learning in Python"*. The Journal of machine Learning research, 12, 2825-2830.

[76] Paszke, A., Gross, S., Chintala, S., Chanan, G., Yang, E., DeVito, Z., ... & Lerer, A. (2017). *Automatic differentiation in pytorch.*